DATA I/O

LogicPak™

950-1942-001

MAR 83 REV A

10-950-1942

Copyright © Data I/O Corporation, 1983. All rights reserved.

Data I/O has made every attempt to ensure that the information in this document is accurate and complete. However, Data I/O assumes no liability for errors, or for any damages that result from use of this document or the equipment that it accompanies.

Data I/O reserves the right to make changes to this document without notice at any time.

NOTE

Before using the LogicPak™, read section 1 for programmer mainframe compatibility information.
If you are not familiar with programmable logic device technology, it may be helpful for you to read the glossary in appendix B to familiarize yourself with the terminology before reading any of the system documentation.

Digitized by the Internet Archive
in 2025

https://archive.org/details/manualsplus_09765

TABLE OF CONTENTS

SECTION 1. INTRODUCTION

1.1 OVERVIEW 1-1

1.2 PLDS SYSTEM OVERVIEW 1-2

1.3 PROGRAMMER COMPATIBILITY 1-3

1.4 THEORY OF OPERATION 1-4
1.4.1 Data Development 1-4
1.4.2 Programming 1-5
1.4.3 Functional Testing 1-6

1.5 LOGICPAK™ OVERVIEW 1-9

1.6 APPLICATIONS 1-9

1.7 SPECIFICATIONS 1-9

1.8 FIELD APPLICATIONS SUPPORT 1-9

1.9 WARRANTY 1-9

1.10 SERVICE 1-9

1.11 ORDERING 1-9

SECTION 2. INSTALLATION

2.1 INSPECTION 2-1

2.2 LOGICPAK™ INSTALLATION 2-1

2.3 LOGICPAK™ REMOVAL 2-2

2.4 ADAPTER INSTALLATION 2-2

2.5 ADAPTER REMOVAL 2-2

2.6 SERIAL INTERFACING 2-3

2.7 REPACKING FOR SHIPMENT 2-3

SECTION 3. OPERATION

3.1 OVERVIEW 3-1

3.2 POWER UP 3-1

3.3 POWER DOWN 3-3

3.4 BASIC DATA TRANSFER OPERATIONS 3-3
3.4.1 Family Code and Pinout Selection 3-3
3.4.2 Device Insertion 3-4
3.4.3 Device Removal 3-4
3.4.4 Load RAM With Master Device Data 3-4
3.4.5 Program Device With RAM Data 3-5
3.4.6 Verify and Functionally Test Device 3-6

3.5 SYSTEM COMMANDS 3-7
3.5.1 Enable Terminal Mode 3-7
3.5.2 Display Command Menu 3-10
3.5.3 Family Code and Pinout Code 3-11
3.5.4 Set Reject Count Option 3-11
3.5.5 Select Verify Option 3-12
3.5.6 Select Security Fuse Option 3-12
3.5.7 Set Number of Logic Fingerprint™ Test Cycles 3-13
3.5.8 Enter Logic Fingerprint™ Test Starting Vector 3-14
3.5.9 Enter Structured Test Vectors 3-15
3.5.10 Display Fuse Pattern 3-16
3.5.11 Receive Fuse Data 3-17
3.5.12 Transmit Fuse Data 3-17
3.5.13 Display Sum-Check of Fuse Data 3-17
3.5.14 Enter Fuse Number and State 3-18
3.5.15 Display Configuration Number 3-20
3.5.16 Exit Command 3-20

SECTION 4. MAINTENANCE/CALIBRATION/TROUBLESHOOTING

4.1 OVERVIEW 4-1

4.2 MAINTENANCE 4-1
4.2.1 Cleaning 4-1
4.2.2 Inspection 4-1

4.3 CALIBRATION 4-1

4.4 TROUBLESHOOTING 4-5
4.4.1 Equipment Set Up 4-5
4.4.2 No System Operation 4-6
4.4.3 LogicPak™ Failure 4-7
4.4.4 LogicPak™ Assembly 4-7

SECTION 5. CIRCUIT DESCRIPTION

5.1 INTRODUCTION 5-1

5.2 GENERAL ARCHITECTURE 5-1

5.3 COMPONENT LAYOUT 5-1
5.3.1 Motherboard 5-1
5.3.2 Extender 5-1
5.3.3 Controller/Memory 5-1
5.3.4 Waveform Generator 5-4
5.3.5 Comparator/Rise Time Board 5-4
5.3.6 Pin Driver 5-4
5.3.7 Sink Driver 5-9

APPENDIX A. Standard Data Transfer Format Between Data Preparation System and Programmable Logic Device Programmer A-1

APPENDIX B. Reference Material B-1

LIST OF FIGURES

1-1 Programmable Logic Development System Components 1-2
1-2 Example of a Modification Status Sticker 1-3
1-3 Sample Sum-Check Calculation 1-5
1-4 Verifying and Testing Procedure Flowchart 1-6
1-5 Logic Fingerprint™ Test Block Diagram 1-8
2-1 LogicPak™ Installation 2-1
2-2 Adapter Installation 2-2
2-3 Sample Interconnection Methods 2-3
3-1 Loading Data Via the Serial Port or Master Device 3-1
3-2 Fuse Map Applied to Logic Device Diagram 3-2
3-3 Programmer Power Switch Location 3-3
3-4 Device Installation 3-4
3-5 Operational Overview Flowchart 3-8
3-6 Complete Fuse Pattern 3-16
3-7 Complete Logic Diagram 3-16
3-8 Sample Printout of JEDEC Format 3-17
3-9 Logic Diagram 3-19
4-1 LogicPak™ Cover Removal 4-1
4-2 Calibration Equipment Setup 4-2
4-3 LogicPak™ Test and Adjustment Locations 4-2
4-4 LogicPak™ Disassembly 4-5
4-5 Troubleshooting Equipment Setup 4-6
5-1 Principal Components of LogicPak™ 5-1
5-2 LogicPak™ Electronics Block Diagram 5-2
5-3 Controller/Memory (701-1942) Block Diagram 5-3
5-4 Waveform Generator (701-1939) Block Diagram 5-5
5-5 Comparator/Rise Time (701-1941) Block Diagram 5-6
5-6 Pin Driver (702-1943) Block Diagram 5-8
5-7 Sink Driver (701-1940) Block Diagram 5-10

LIST OF TABLES

1-1 Using the LogicPak™ Manual 1-1
1-2 Programmer Compatibility 1-4
1-3 RAM Capacity for Structured Test Vectors 1-7
3-1 PLDS System Command Summary 3-9
4-1 LogicPak™ Error Codes 4-3

SECTION 1

INTRODUCTION

1.1 OVERVIEW

Data I/O's Programmable Logic Development System (PLDS) provides data development, programming, and testing support for logic device families from most semiconductor manufacturers. Once the programmer data RAM (random access memory) has been loaded with a specific fuse pattern, devices can be programmed by installing the appropriate adapter in the PLDS. With the PLDS, system designers can now use one integrated system to specify logic device functions using Boolean equations or truth tables, can program, and can functionally test devices.

This manual describes the components and operation of the PLDS. Subjects addressed in this manual and the corresponding sections are listed in Table 1-1. Use this list as a quick reference point of the major sections in this manual.

Throughout this manual, the entries that you are to make from either the programmer or terminal are indicated by the entry enclosed in a key symbol. For example,

indicates the "ESCAPE" key on the terminal keyboard should be pressed. In addition, the symbols shown below are used to indicate modes of operation and prompts.

29A TERMINAL

Table 1-1. Using the LogicPak™ Manual

Subject	Section
Interrelationships of system components	1.2
Functions of system components	1.2
Conceptual overview of data development	1.4
Conceptual overview of programming	1.4
Conceptual overview of testing operations	1.4
Installation procedures for LogicPak™ and adapter installation	2.0
Basic operation instructions for LogicPak™ with the 29A Universal Programmer	3.0
Exact key sequences for operating LogicPak™ with the 29A Universal Programmer (refer to your programmer manual for System 19 and 100A key sequences)	3.0
Maintenance for LogicPak™	4.0
Calibration for LogicPak™	4.0
Troubleshooting for adapters (also in each appropriate adapter manual)	4.0
Circuit descriptions for LogicPak™ (in appropriate adapter manual for the adapters)	5.0
Block diagram for LogicPak™ (in appropriate adapter manual for the adapters)	5.0
Schematics for LogicPak™ (in appropriate adapter manual for the adapters)	5.0
Transfer of information between data preparation system and a device programmer	Appendix A
Glossary	Appendix B
Logic schematics	Appendix B
Data I/O Service Centers	Back of manual
Warranty information	Back of manual

TM LogicPak is a trademark of Data I/O.

1.2 PLDS SYSTEM OVERVIEW

The PLDS provides a universal means for developing data, programming, and testing a variety of logic devices in engineering and production environments. This modular system comprises the LogicPak™, a programmer, adapters, and a terminal (see figure 1-1).

Like the Data I/O programming paks, the LogicPak™ is designed to interface with a standard programmer such as the 29A Universal Programmer (see section 1.3 for programmer compatibility). The LogicPak™ contains the firmware and electronics necessary for PLDS operations and controls the flow of data to and from the user as well as throughout the system. In this manual we will refer to the operational procedures for the 29A Universal Programmer; refer to your programmer manual for System 19 and 100A production programmer key sequences.

Software tables resident within the LogicPak™ store values for programming variables, including pinouts, voltage levels, and timing. When you choose the family and pinout codes for a particular device, the programmer uses information in these tables to assemble a specialized programming routine in scratch RAM. This allows high-speed operation with minimum firmware.

The PLDS has two types of adapters: programming/testing (P/T) adapters and design adapters. There is one P/T adapter for each manufacturer's device family. These adapters fit onto the top of the LogicPak™ and provide the supplementary, device-specific software to convert the LogicPak™, programmer, and a terminal into a programmable logic development system. A list of available logic devices and the hardware necessary for data development, programming, and testing of each device is included with each adapter manual and is on the "wall chart" (Data I/O Comparison Chart of Programmable Devices).

As Data I/O increases the capabilities of the LogicPak™ to program new or additional devices, firmware updates will be available for existing adapters to add new devices to existing device families. New adapters may also be added to the PLDS to accommodate new device families.

The PLDS programs a wide variety of logic devices, such as FPLA (field programmable logic array), PAL® (programmable array logic), FPLS (field programmable logic sequencer), and FPGA (field programmable gate array), with a minimum of equipment changes. Because P/T adapters configure the LogicPak™ to specific

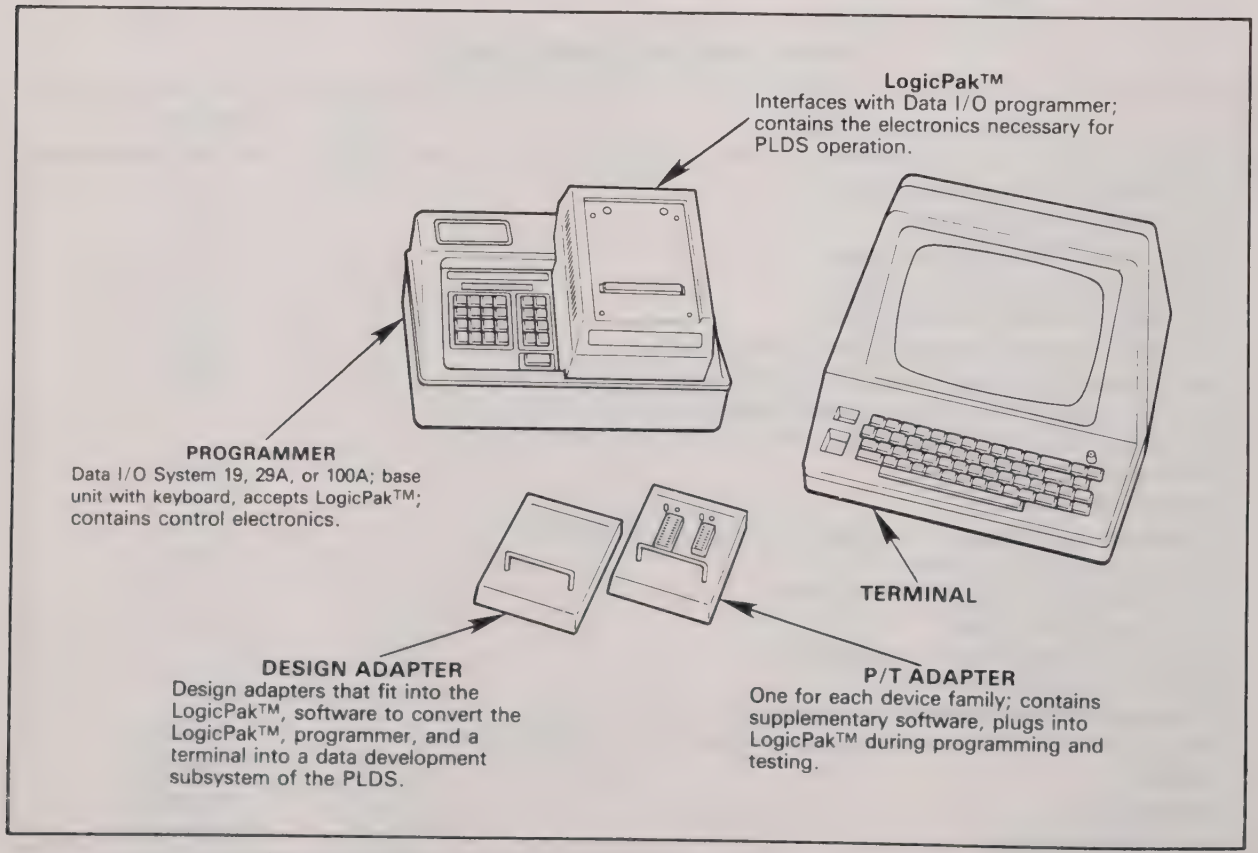

LogicPak™
Interfaces with Data I/O programmer; contains the electronics necessary for PLDS operation.
PROGRAMMER
Data I/O System 19, 29A, or 100A; base unit with keyboard, accepts LogicPak™; contains control electronics.
TERMINAL
DESIGN ADAPTER
Design adapters that fit into the LogicPak™, software to convert the LogicPak™, programmer, and a terminal into a data development subsystem of the PLDS.
P/T ADAPTER
One for each device family; contains supplementary software, plugs into LogicPak™ during programming and testing.

Figure 1-1. Programmable Logic Development System Components

® PAL is a registered trademark of Monolithic Memories, Inc.
™ Logic Fingerprint test is a trademark of Data I/O

manufacturers' device families, the only hardware you have to change when you change device families is the P/T adapter.

Programmed devices can be functionally tested using an optional method developed by Data I/O called the Logic Fingerprint™ test. This unique new test, which can be used for combinational and sequential devices, finds defective devices that could pass a routine fuse verify operation. The Logic Fingerprint™ test is fast and easy and does not require you to generate test vectors. See section 1.4.3 for a full description of this feature.

A structured test can also be performed on programmed devices, based on the test vectors you generate. This additional testing usually will not be necessary with most combinational devices because the Logic Fingerprint™ tests them adequately. With some sequential devices, the Logic Fingerprint™ test cannot test all active states. It checks a statistically significant sample, but not necessarily a specific state you desire. When it is necessary to define certain test sequences explicitly, the structured test allows you to write test vectors and test specific states for full assurance of proper functionality.

These test vectors are applied to the device in sequence and apply desired inputs and check for correct outputs. When used in conjunction with the Logic Fingerprint™ test, they greatly increase the likelihood of finding defective devices. When entered, structured tests are always performed prior to the Logic Fingerprint™ test. Device power remains applied between structured testing and Logic Fingerprint™ testing, therefore allowing initialization of sequential devices. Some sequential devices require structured test vectors and/or selection of the Logic Fingerprint™ test starting vector to avoid false errors when performing the Logic Fingerprint™ test. These requirements are device and fuse-pattern dependant; refer to the appropriate P/T adapter manual for specific examples of the devices that are subject to special considerations.

New data development features simplify design efforts. Monolithic Memories' PALASM (PAL assembler) and Signetics H&L (high-and-low) logic development software are contained in the design adapters.

PALASM is a development language that allows you to describe the desired operation of a PAL device using Boolean equations. Signetics H&L is used to design IFL (integrated fuse logic) devices using logic function tables.

1.3 PROGRAMMER COMPATIBILITY

In addition to the 29A, Data I/O's System 19 and 100A Production Programmers are compatible with the LogicPak™. To be compatible with the LogicPak™, your programmer may require a hardware and/or firmware update, depending on the model, configuration, and age. The following information will help you determine if your programmer requires updating. If you find that your programmer does require updating, contact your nearest Data I/O Service Center. (A list of Service Centers is at the back of this manual.)

- System 17—The System 17 must be converted into a System 19 with the latest firmware installed and with the latest hardware modifications.
- System 19—Check to determine if your System 19 contains a 702-1520 or 702-1980 controller board by performing the following steps:
  1. Remove the programming pak.
  2. Remove the metal shield (if any); see section 2.2, figure 2-1.
  3. Count the number of EPROM firmware sockets located just behind the pak interface connector. If there are four sockets, it is a 702-1520 board. If there are eight sockets, it is a 702-1980 board.

If your System 19 contains a 702-1520 controller board, check the modification status sticker (figure 1-2) on the bottom of the programmer. If the sticker is not there, or if only "1" is marked off, your System 19 requires hardware and firmware updating. Contact the nearest Data I/O Service Center. If "2" is marked, your System 19 is compatible with the LogicPak™. If your System 19 contains a 702-1980 controller board, it may require a firmware update. To display the configuration number of the firmware in your programmer, press

If the configuration number displayed is "3599" your firmware needs updating.

MODIFICATION STATUS

1	2	3	4	5	6
●	0	0	0	0	0

Figure 1-2. Example of a Modification Status Sticker

- 29A Universal Programmer—To be compatible with the LogicPak™ the 29A programmers must have Rev (revision) C or later firmware. To determine the configuration of the firmware in your 29A, press

and observe the display. If the hexadecimal number matches one listed in table 1-2, your firmware needs to be updated.

Table 1-2. Programmer Compatibility

Model	Revision	Configuration Number
29A	A B	1ECA 20A4
29A (with computer remote control)	A B	BB41 C00B
100A	A B C D	917F 9405 9DEE 9BED

- 100A Production Programmer—To be compatible with the LogicPak™, the 100A programmers must have Rev E or later firmware. To determine the configuration of the firmware in your 100A, press

and observe the display. If the hexadecimal number display matches one listed in table 1-2, your firmware needs to be updated.

1.4 THEORY OF OPERATION

Logic devices are programmed by first defining the logic structure of a programmable logic device. This first step is known as data development. The logic structure is specified using a function table or a set of Boolean equations. These Boolean or function table designs must be translated into a fuse pattern in RAM using a design adapter.

Once the fuse pattern is translated, you can program the device using a P/T adapter. The programmer can then blow the proper fuses to implement the logic design. As simple as this process sounds, many programmers require manual translation of the logic design, akin to manually translating assembly language software into machine language. The programming steps are virtually the same as those taken when programming PROMs. To start the programming sequence, first enter a device code, then programming can begin. (See section 3.4.)

When a manufacturer ships a programmable logic device, the manufacturing cycle is only partially complete. Because the device is not fully manufactured until it is programmed, the fault coverage of the manufacturer's final test is limited. Therefore, the programmer must be able to perform some type of functional testing. This is accomplished in the PLDS by using structure testing and Logic Fingerprint™ testing. Refer to section 1.4.3 for additional information.

1.4.1 DATA DEVELOPMENT

The first step in data development is to design the logic using some convenient language. The PLDS contains two adapters to enable you to develop your logic design: PALASM design adapter and H&L design adapter. The PALASM adapter is to be used with PAL-type devices and translates Boolean equations into fuse data. The H&L design adapter is used with IFL-type devices and translates logic function tables into fuse data.

Those design adapters are used in conjunction with a terminal that is connected to the programmer. The PLDS enables you to specify a device and input the design in the form of Boolean equations or logic function tables from the terminal via RS-232 port in the programmer.

The adapter firmware translates your design into a fuse pattern and, optionally, test vectors. This fuse pattern and test data are resident in the programmer's RAM. Thus, the design adapter can be removed from the LogicPak™, and an appropriate P/T adapter can be installed allowing the fuse pattern to be programmed directly into the device and automatically tested. The detailed operating procedures for data development are provided in the PALASM and H&L design adapter manuals.

If a fuse pattern is generated on a host system, it must use fuse numbers specified according to logic diagrams in the P/T adapter manual and transmitted to the programmer in the JEDEC (Joint Electron Device Engineering Council) format (see appendix A). Data I/O uses the JEDEC Logic Device Translation Format (number JC-42, 1-81-62) for serial data input and output with the LogicPak™. The only exception to this is when you are using a Signetics H&L design adapter, in which case data transfer can also occur in the Signetics H&L logic format.

NOTE

Before operating, see the JEDEC format specifications limitations in appendix A, section 3.0.

An alternate method of specifying the fuse map is to manually enter the fuse number and state for every fuse in the device. Each P/T adapter manual contains logic diagrams for the devices in its repertoire. These are the same as those in the device manufacturers' data books, but the fuse number has been added. Although tedious, fuse numbers and states can be entered manually into the programmer's data RAM from the programmer's keyboard or from a terminal. This method usually will be used only for editing fuse data because it is a long process with room for error.

With a P/T adapter, fuse data can be entered into the programmer's RAM by loading from a master device. Blank devices can then be programmed using the same P/T adapter, or other manufacturers' functionally equivalent second-source devices can be programmed by installing the appropriate P/T adapter. Remember that a device that has its security fuse programmed **cannot** be used as a master because its fuses cannot be read.

A different programmer RAM fuse map was used in some previous Data I/O logic device programming modules (the 950-0800, 919-1427, and 919-1542). If you have paper tapes or files that you prepared to use with these modules, you will have to prepare new files for use with the LogicPak™ unless you used the Signetics H&L logic format. The preferred translator format is the JEDEC format, but a Signetics translator is available. The 950-0104 and 919-0045 modules use a memory map that is compatible with the LogicPak™. The 919-1427 pak used a fuse map generated to simulate a 512 x 4 PROM. Serial data were entered using a standard PROM data translation format. Again, tapes must be regenerated for use with the LogicPak™ because of the different correspondence between fuses and bits in RAM. The fuse maps have been changed to allow for the programming of functional second-source devices from the same fuse map loaded in RAM and to avoid gaps (previously called phantom fuses) in the fuse maps.

NOTE

If master devices are used as a data source, the above considerations do not apply. A master device can be used as the transfer medium between different fuse maps. Use the old pak (old fuse map) to program a master device, then install the LogicPak™ and load the master. The new LogicPak™ fuse map will then reside in RAM.

Sum-Check

After fuse data have been loaded into the PLDS from the serial port or the master device, the programmer calculates the sum-check of the fuse data (figure 1-3) and displays it (when in terminal mode, it will be displayed on the terminal). The sum-check, which is used to verify the integrity of data transfers, is a summation of eight-bit bytes of fuse data expressed as a four-digit hexadecimal number; see figure 1.3.

HEXADECIMAL DATA	BINARY DATA
84	10000100
C1	11000001
62	01100010
24	00100100
01CB	0000 0001 1100 1011
↑ Sum-check in hexadecimal notation	↑ Sixteen-bit binary sum-check

Figure 1-3. Sample Sum-Check Calculation

If data were loaded through the serial port and a sum-check was sent with it, the programmer will compare the sum-check with its own calculation. If they agree, the correct sum-check is displayed. If they do not agree, the programmer will signal an error. Data from the serial port will also be checked for correct parity if the program parity switch is on (see your programmer manual).

Data from a master device are loaded into RAM by installing the correct P/T adapter on the LogicPak™ and performing a load operation. Any information in the source buffer (Boolean equations, function tables, or test vectors) will not be affected.

Editing Features

After data have been developed or loaded into programmer RAM, the PLDS provides editing features that allow you to modify the fuse pattern. These are accessed through select functions as described in section 3.5.

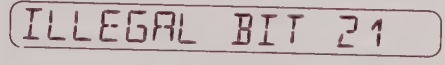
Data Lock

The System 19, 100A, and 29A programmers have a data-lock feature that allows you to limit the accessibility of your RAM data. This is useful in production environments to protect the integrity of your RAM data. Refer to the programmer manual for information on data lock.

1.4.2 PROGRAMMING

After data development, the next step in the programming sequence is programming the logic devices. The LogicPak™ applies the manufacturer's specific algorithms to blow fuses in the logic device according to the fuse pattern in the programmer RAM. Once you key in the operation on the programmer, programming is automatic and starts with a series of tests: backward device test, illegal bit test, and blank check. During the backward device test, the programmer automatically checks the device's orientation in the P/T adapter socket and displays an error if it is inserted backwards,
the 29A displays:

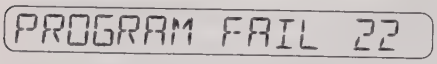
The illegal bit test checks for previously programmed bits in a nonblank device that cannot be programmed according to the fuse pattern in RAM. If illegal bits exist,
the 29A displays:



During the blank check, the programmer searches the device for programmed bits. If any are found (and the bits are legal), the programmer will signal the operator,
the 29A displays:

Nonblank parts can be over-programmed by again pressing

If the device passes these tests, data are transferred from the programmer RAM to the LogicPak™ one byte (eight fuse states) at a time. The LogicPak™ then applies the programming pulses to the first fuse and tests its condition. If the fuse fails to program, the 29A displays:



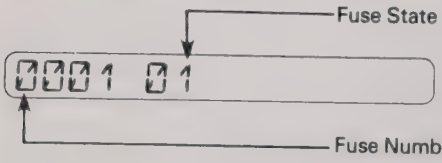
Otherwise, programming proceeds to the next fuse until all have been programmed. (With some P/T adapters, programming algorithms vary and may only display a verify error.) Refer to the P/T adapter manuals for manufacturer-specific programming algorithms and waveform pictures.

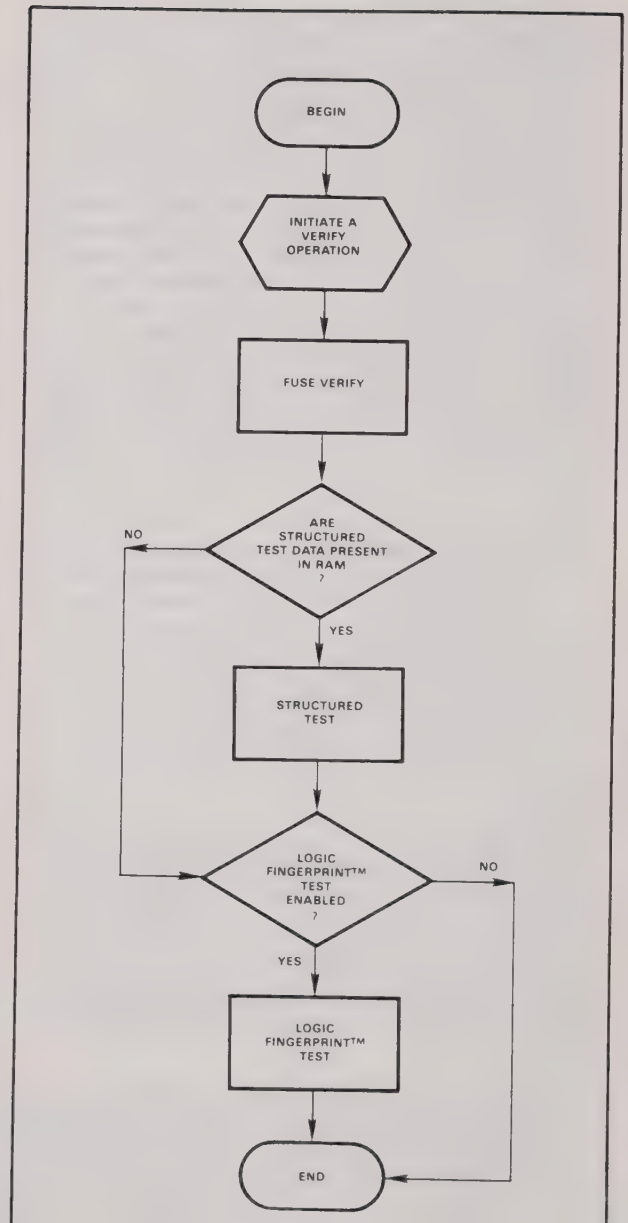
1.4.3 FUNCTIONAL TESTING

The next step in the programming sequence is functional testing, which verifies that the device will perform as intended. Figure 1-4 shows the sequence used to verify and test devices with the LogicPak™; see section 3.4.5 for details on the verify options.

Fuse Verify. The fuse verify ensures that the fuse pattern in the device and the programmer RAM are the same, which would indicate that all fuses have been correctly programmed. If the fuse pattern of the device and the programmer RAM correspond, the verify operation will display the sum-check of the RAM data.

During a fuse verify, the LogicPak™ compares the fuse pattern of the programmed device, one fuse at a time, with the fuse pattern in the programmer data RAM. When fuse states in RAM do not correspond to those in the programmed device, the PLDS will display the fuse number, the fuse state in RAM, and whether the first or second pass verify produced the failure. For example, the 29A will display:





NOTES

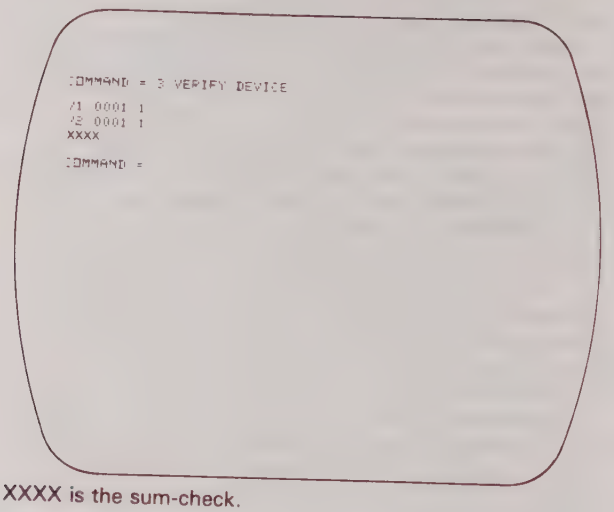
1. *This sequence will occur automatically after a program operation. To verify a previously programmed device, the VERIFY key on the programmer keyboard or a remote control verify command is used.*

2. *The structured test occurs (if data are present) before Logic Fingerprint™ test. This is useful to initialize sequential devices so the Logic Fingerprint™ test is more effective.*

Figure 1-4. Verifying and Testing Procedure Flowchart

ca
(b

The terminal will display:



XXXX is the sum-check.

Most of the logic paths of a programmable logic device are **not** tested by the fuse verify. Therefore, a fuse verify will not guarantee that the device will perform its intended function; however, it is a necessary step to ascertain whether or not the device has been programmed correctly.

Structured Test. The structured test is optional. A structured test will be performed on the device automatically after programming and during a verify operation only if there are structured test data present in the programmer data RAM. If no data are present, only a fuse verify and a Logic Fingerprint™ test (if enabled) will be performed.

The structured test supplements the Logic Fingerprint™ test. It lets you enter test vectors that stimulate and read device pins, guaranteeing that specific states will be tested. It can also be used to initialize devices for the Logic Fingerprint™ test. The structured test enables you to uniquely define the inputs and test for desired outputs. The LogicPak™ applies those inputs and verifies that the desired outputs appear. The structured test cannot be performed unless there are structured test vectors in RAM. With the vectors present, the structured test can be disabled by selecting verify option "1" (see section 3.4.5).

While in the terminal mode, when a device fails a structured test, the terminal will display the vector test number and output pin number that failed to show the specified levels. When using the programmer, the 29A will display an error code. (See section 4, table 4-1.) The structured test requires that you write your own test vectors and input them to the programmer RAM, or that you use data development aids to generate the test vectors from specified inputs, outputs, or function tables. Although a structured test guarantees that certain specified states have been tested, writing vectors is a time-consuming task that requires a thorough understanding of the designed function of the device under test.

Test vectors can be developed efficiently with the PALASM design adapter. This design aid allows you to enter equations and function tables. Firmware compares the equations and function tables during an operation called "Simulate Source Equations." If the equation and function tables are valid, the firmware generates structured test vectors. Data I/O recommends this method as it is the fastest and easiest way to produce structured test vectors. If vectors are developed in any other way, they must be manually input to the programmer RAM from the terminal.

The structured test requires you to write your test vectors and input them to the programmer data RAM. The programmer RAM is limited as to the number of tests it can store. See table 1-3. Each vector tests one specific state; for example, if RAM is limited to 50 states and you require testing 100 unique states, you would test 50 states at one time and use multiple loads and test sequences.

Table 1-3. RAM Capacity for Structured Test Vectors

Device Pins	Structured Tests
20	50
24	42
28	36

The structured test can be used to: 1) apply all possible input states to a combinational device, 2) force a sequential device through all its state transitions, and 3) because the structured test is performed before the Logic Fingerprint™ test, it can be used to present a series of inputs to a device that will drive it to a known initial state so the Logic Fingerprint™ test can begin from that known state. This is required for registered or sequential devices that may power up in an illegal state. Structured testing is especially useful with sequential devices because pseudorandom testing algorithms do not achieve the same level of test coverage as they do for combinational devices.

Some manufacturers' sequential devices include a preload feature that enables the device to be preset to a known initial state. This preload feature is enabled by specifying special variables in the structured test vector. Refer to section 3.5.9 for specific structured test vectors. Sequential devices without this feature will need to be designed and programmed so that structured vectors can force them into a known initial state. This means that you must design the registered (sequential) device so that it can be tested, then write structured test vectors that initialize the device for testing.

Once written, structured vectors can be stored as part of the JEDEC formatted file (see appendix A).

Logic Fingerprint™ Test. Of all testing methods available in logic device programmers, the Data I/O Logic Fingerprint™ test is the easiest to use and is flexible enough

to be applicable to a most logic devices. The Logic Fingerprint™ test is similar to signature analysis. To test a device using the Logic Fingerprint™ test, it is necessary for the LogicPak™ to first learn a signature (test-sum) from a known-good device. Subsequently programmed devices will be tested, and their test-sums will be compared against the reference test-sum. If the test-sums match, the device has passed the Logic Fingerprint™ test.

Figure 1-5 shows a block diagram of the Logic Fingerprint™ test. The test uses a shift register as a pseudorandom pattern generator to stimulate the device under test with test vectors. Depending on how the device is programmed, it responds to these inputs with some set of outputs. These outputs are used in conjunction with the inputs to generate the next vector.

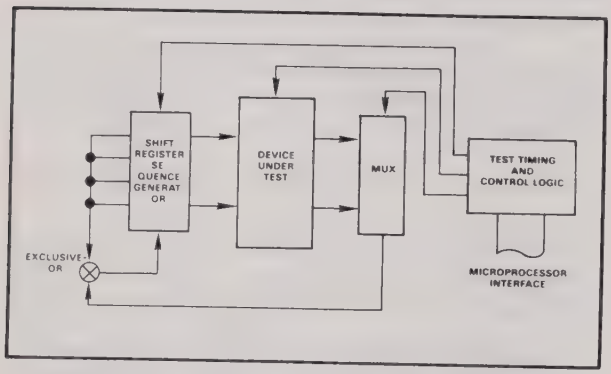


Figure 1-5. Logic Fingerprint™ Test Block Diagram

The outputs are fed back throughout the pseudo-random sequence generator, 128,000 times for each cycle of the Logic Fingerprint™ test. The contents of the register after the test is the test-sum unique to the programmed device. The test-sum is then automatically compared to the stored reference test-sum.

The LogicPak™ learns the reference test-sum from a known-good master device during a load operation. If already known, it can be loaded via the serial port along with programming data (in the JEDEC format) or entered from the programmer keyboard or terminal. If a structured test is being used to initialize devices for the Logic Fingerprint™ test, the structured test will be performed upon a load operation prior to learning the test-sum. If the device fails the structured test in load, an error code will be displayed (see section 4, table 4-1).

The Logic Fingerprint™ test is enabled by entering the number of test cycles desired. The default number of cycles is zero, which disables the test. Any number of cycles up to 99 can be entered; however, each additional cycle will increase the time of the Logic Fingerprint™ test. Each cycle takes 3 to 6 seconds, depending on the device type. Only 1 to 8 cycles are necessary in most cases. More than one cycle will be used most often with sequential devices that require more than 128,000 testing cycles for a satisfactory test.

The operator can specify the starting vector for the Logic Fingerprint™ test. The starting vector could be used in several instances: 1) to initialize a sequential device that may not power-on reliably to a known initial state or 2) to test a device that will not respond to the default starting vector of all bits set to zero. Refer to the P/T adapter manuals for examples. The starting vector will enable the Logic Fingerprint™ test to begin from a known initial state. The starting vector can be up to 28 bits long, depending on the number of pins on the device. A 20-pin device will have a 20-bit starting vector, and a 24-pin device, 24-bit vector.

The pseudorandom nature of the input vectors can cause some devices in some programming circumstances to fail the Logic Fingerprint™ test by giving nonrepetitive test-sums. THIS DOES NOT NECESSARILY INDICATE A FAULTY DEVICE, but may be an indication that the device is subject to the Logic Fingerprint™ test limitations. Logic Fingerprint™ test limitations are described in each P/T adapter manual for each manufacturer's devices. IT IS VERY IMPORTANT THAT YOU READ AND UNDERSTAND THESE LIMITATIONS.

Logic Fingerprint™ test can be disabled by setting the number of Logic Fingerprint™ test cycles at zero. (This is the normal default.) If Logic Fingerprint™ test is not suitable for a specific device, it can always be tested by using the structured test.

Self-Testing. The LogicPak™ performs a series of self-tests on its hardware to assure the functionality of the module. A failure will display an error code; see section 4, table 4-1. To isolate the problem to its source, refer to the troubleshooting section (section 4.4).

Because of the internal analog feedback capability in the LogicPak™, voltage levels can be measured so that a go/no-go test can be performed. Programmable voltage levels are tested when device-related operations are executed. The TTL (transistor-transistor logic) levels are tested when the family and pinout codes with the system 29A and 100A programmers and when the device-related operations are executed with the system 19 programmer. Testing is accomplished under software control by first setting test levels and then reading back these levels through analog comparators. The comparator output levels are then verified against predetermined values.

When the programmer is powered up, the Logic Fingerprint™ self-test is done by testing an open socket. The LogicPak™ compares the open-socket test-sum against a predetermined test-sum stored in the firmware. This guarantees that all Logic Fingerprint™ test circuitry is functional. These tests are performed during calibration operations; see section 4 for the diagnostic steps.

The self-test routines will pinpoint any failures immediately and flag with an error if there is a problem. Therefore, you can be confident that your equipment is in proper operating condition. All the testing features Data I/O has incorporated into the PLDS give you the security of knowing you will have the highest possible programming yields. The hardware self-tests assure you your equipment is

capable of programming and testing. The device test (backwards device test, illegal bit, blank check, fuse verify, Logic Fingerprint™ test, and structured test) assure you that your devices are programmed correctly and test functionally. Together these tests result in the maximum programming yield possible with any commercial programmer.

1.5 LOGICPAK™ OVERVIEW

The Data I/O 303A LogicPak™ is the base unit of the PLDS. It contains all the common electronics and firmware for programming and testing logic devices. Any electronics or firmware unique to a specific device family, or device within a family, are resident in P/T adapters that plug into the LogicPak™. Families with more than one pin number series (e.g., PAL 20 and PAL 24) have sockets to accommodate each pin count.

Software tables store values for programming variables including pinouts, voltage levels, and timing. When you choose the family and pinout codes for a particular device, the programmer uses information in these tables to assemble a specialized programming routine in scratch RAM. This method allows high-speed operation with minimum firmware overhead.

To increase flexibility in waveform generation, digital-to-analog converters (DAC) control all major power supplies, with several rise and fall times selected by software.

In addition to its programming and testing capabilities, the LogicPak™ is the base unit for design adapters. These adapters permit high-level Boolean equation or function-table entry.

1.6 APPLICATIONS

A complete list of all currently available logic devices and the hardware necessary for data development, programming, and testing of each is included with each adapter manual. As Data I/O increases the capabilities of the LogicPak™ to program new or additional devices, firmware updates will be available for existing adapters to add new devices to existing device families. New adapters may also be added to the PLDS to accommodate new device families.

1.7 SPECIFICATIONS

The LogicPak™ receives its power from the programmer power supplies. Programming waveforms are generated from programmer supplies using DACs controlled by the programmer's microprocessor. The controlling firmware is located both on a circuit board in the LogicPak™ and in the P/T adapters or design adapters. The physical and environmental specifications of the LogicPak™ are:

- altitude (operating): sea level to 3 km (10,000 ft)
- humidity (operating): 90% maximum (noncondensing)
- humidity (storage): 95% maximum (noncondensing)
- temperature (operating): −5 to 45°C (41 to 113°F)
- temperature (storage): −40 to 70°C (−40 to 158°F)
- weight: 1.6 kg (3 lb, 8 oz)
- dimensions: 17.9 x 17.3 x 21.7 cm (7.05 x 6.81 x 8.54 in.)

1.8 FIELD APPLICATIONS SUPPORT

Data I/O has field applications engineers throughout the world. They can provide additional information about interfacing Data I/O products with other systems and answer questions about your equipment.

These engineers are located within the United States at the addresses listed in the back of this manual. For international applications support, contact your nearest Data I/O representative.

1.9 WARRANTY

The Data I/O LogicPak™ is warranted against defects in materials and workmanship. The warranty period of 90 days begins when you receive the equipment; the warranty card inside the back cover of this manual explains the length and conditions of the warranty. For warranty service, contact your nearest Data I/O Service Center.

1.10 SERVICE

Data I/O maintains service centers throughout the world, each staffed with factory-trained technicians to provide prompt, quality service. A list of all service centers is located in the back of this manual.

1.11 ORDERING

To place an order for equipment, contact your Data I/O sales representative. Orders for shipment must include:

- a description of the equipment (see the latest Data I/O price list or contact your sales representative for equipment and part numbers)
- purchase order number
- desired method of shipment
- quantity of each item ordered
- shipping and billing address of the firm, including ZIP code
- name of person ordering the equipment.

SECTION 2

INSTALLATION

2.1 INSPECTION

The 303A LogicPak™ was tested both electrically and mechanically before it was shipped and was carefully packaged to prevent shipping damage. It should, therefore, arrive free of any defect, without marks or scratches, and in perfect operating condition. Carefully inspect the instrument for any damage that may have occurred in transit; if you note any damage, file a claim with the carrier and notify Data I/O. Also, check that the following equipment is present:

- 303A LogicPak™ (part number 950-1942)
- Instruction Manual (part number 10-950-1942).

2.2 LOGICPAK™ INSTALLATION

The LogicPak™ may be installed and removed with the programmer's power on; this feature allows you to retain data in RAM during module changes. If the programmer power is turned on before the LogicPak™ is installed, you will hear a beep until the LogicPak™ with an adapter is installed.

NOTE

Voltage transients can cause device damage. If a P/T (programming/testing) adapter is installed in the LogicPak™, be sure that all sockets are empty when:

- *switching power on or off*
- *installing or removing the LogicPak™ or adapter.*

To install the LogicPak™:

1. Slide the LogicPak™ into the opening in the programmer (figure 2-1).
2. Tilt the LogicPak™ up and gently push it back to hook the flange of the LogicPak™ over the back edge of the programmer opening (figure 2-1a).
3. Lower the LogicPak™ into position as shown in figure 2-1b.
4. Press gently on the front edge of the LogicPak™ to ensure a good connection (figure 2-1c).

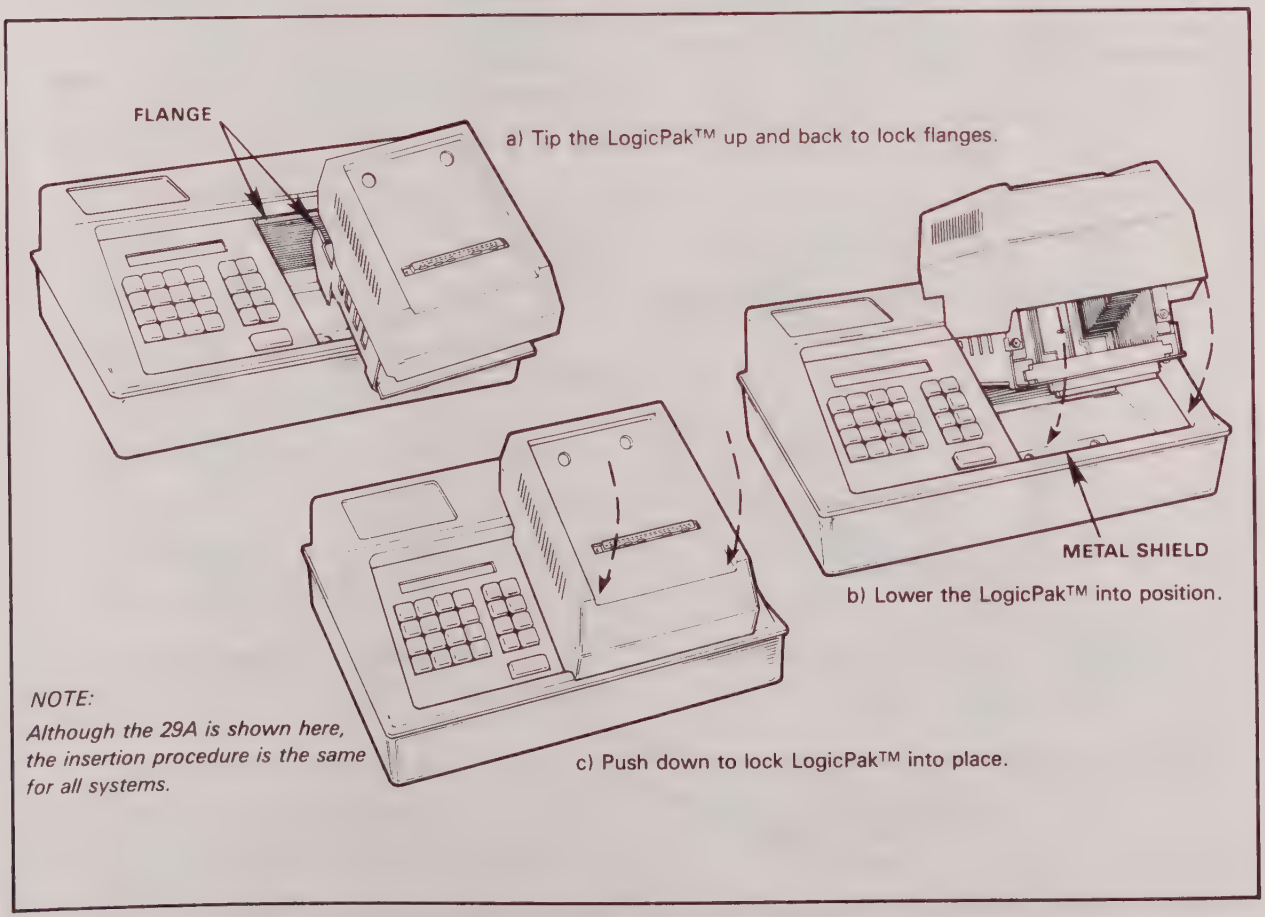

FLANGE
a) Tip the LogicPak™ up and back to lock flanges.
METAL SHIELD
b) Lower the LogicPak™ into position.
NOTE:
Although the 29A is shown here, the insertion procedure is the same for all systems.
c) Push down to lock LogicPak™ into place.

Figure 2-1. LogicPak™ Installation

2.3 LOGICPAK™ REMOVAL

To remove the LogicPak™:

1. Check to make sure the programmer is not in the process of an operation. If it is, wait until the operation is complete; the action symbol on the 29A programmer display disappears (see your programmer manual). In the terminal mode, press ESC (escape).
2. Check to make sure a device is not in a socket. If one is in a socket, remove it as described in section 3.4.2.
3. Grasp the sides of the LogicPak™ and lift.
4. Pivot the LogicPak™ toward the rear of the programmer to unlock the flanges, then lift out.

2.4 ADAPTER INSTALLATION

To insert all adapters for the LogicPak™:

1. Check to make sure a device is not in a socket. If a device is in a socket, remove it as described in section 3.4.2.
2. Align the guide pins on the underside of the adapter with the guide pin holes on the LogicPak™ (see figure 2-2).
3. Gently set the adapter on the LogicPak™. Make sure the guide pins line up with the guide pin holes.
4. Firmly press down on the front edge of the adapter to lock the connector pins into the connector receptacle (see figure 2-2).

2.5 ADAPTER REMOVAL

CAUTION

BEFORE REMOVING THE ADAPTER, press ESC from the terminal, or from the programmer front panel, press the KEYBOARD key (on the System 19) or the VERIFY key (on the 100A or 29A). Because the processor in the programmer executes firmware resident in the adapter, these precautions must be taken before removing the adapter from the LogicPak™ to prevent a program interrupt or loss of RAM data.

To remove the adapter:

1. Ensure that the programmer has completed the current operation.
2. Ensure that a device is not in a socket.
3. While holding down the LogicPak™, grasp the handle and gently remove the adapter.

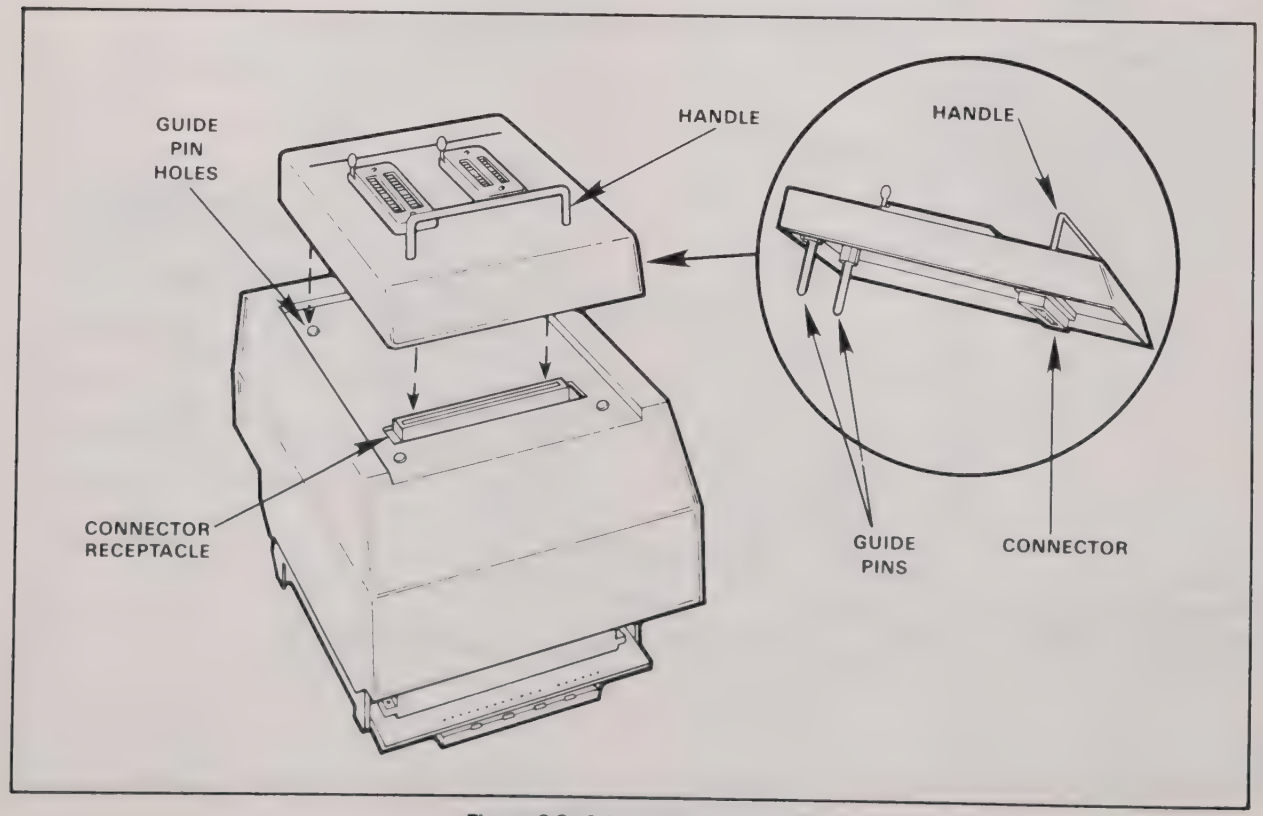


Figure 2-2. Adapter Installation

2.6 SERIAL INTERFACING

The programmer uses an RS-232C serial interface; see figure 2-3. To set the baud rate, parity, and stop bits, refer to your programmer manual; we recommend a transmission rate of 9600 baud.

NOTE

Be sure the programmer power is off before setting the parity and stop bits.

To set up your terminal:

1. Disable any special terminal functions that use CNTL H, CNTL M, CNTL P, or CNTL Z.
2. Select the cursor mode that deletes the character in the current cursor position (e.g., destructive cursor); this will vary from terminal to terminal.
3. Set alpha characters for uppercase (e.g., CAPS LOCK).
4. Set the line mode of the terminal to operate with Data I/O programmers as follows:
   - Full duplex: System 19 (part numbers 1900, 1901, 1903, 1902 if using front panel control).
   - Half duplex: System 19 (part number 1902 if using remote control), 29A, 100A.

2.7 REPACKING FOR SHIPMENT

If the LogicPak™ is to be shipped to Data I/O for service or repair, attach a tag to it describing the work required and identifying the owner. In correspondence, identify the unit by part number, revision level, and the name of the unit. If the original shipping container is to be used, place the LogicPak™ in the container with the appropriate packing materials, and seal the container with strong tape. If another container is used, be sure that it is a heavy carton, wrapped with heavy paper or plastic; use appropriate packing material, and seal well with strong tape. Mark the container "DELICATE INSTRUMENT" or "FRAGILE."

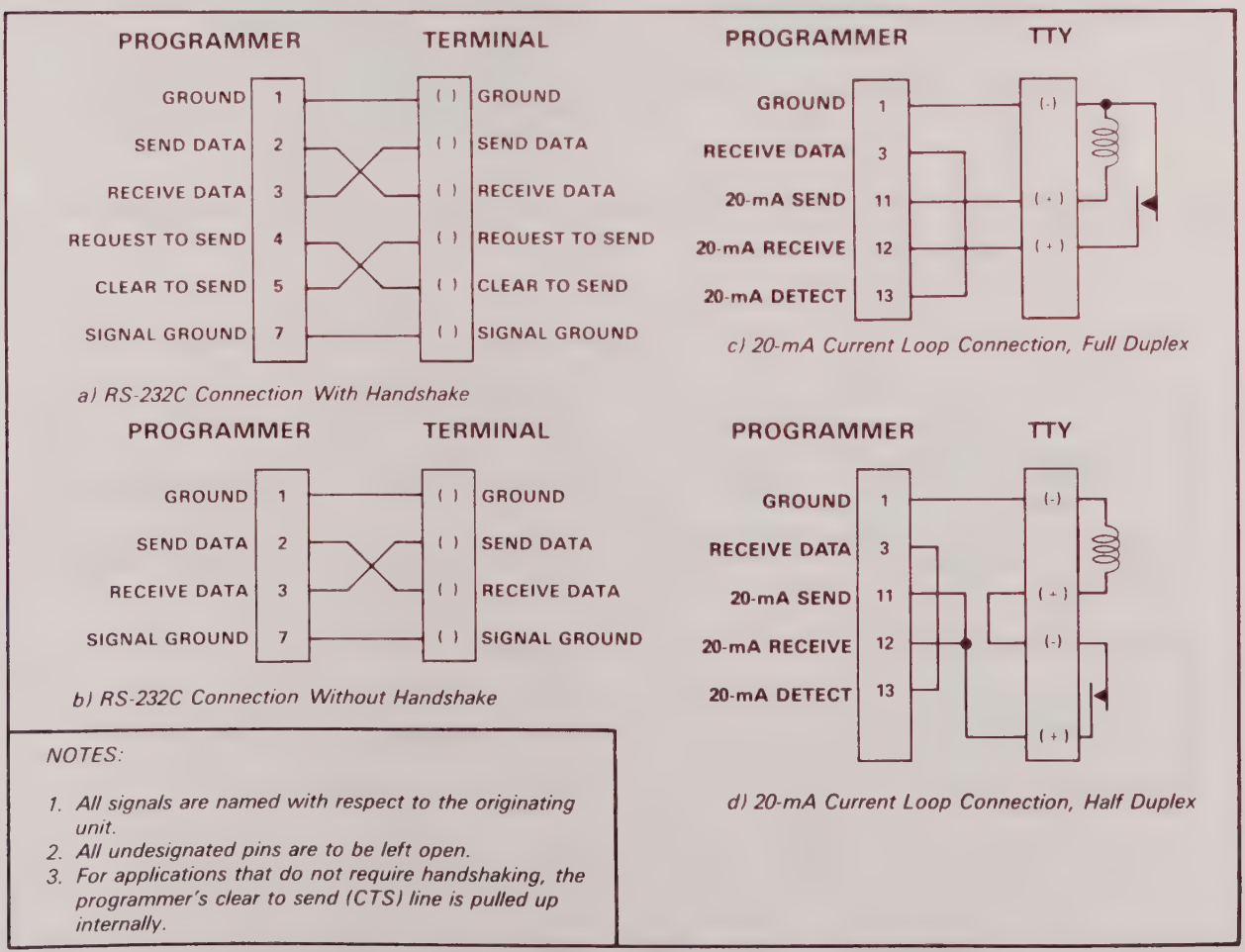

PROGRAMMER
TERMINAL
GROUND 1 ( ) GROUND
SEND DATA 2 ( ) SEND DATA
RECEIVE DATA 3 ( ) RECEIVE DATA
REQUEST TO SEND 4 ( ) REQUEST TO SEND
CLEAR TO SEND 5 ( ) CLEAR TO SEND
SIGNAL GROUND 7 ( ) SIGNAL GROUND
a) RS-232C Connection With Handshake
PROGRAMMER
TERMINAL
GROUND 1 ( ) GROUND
SEND DATA 2 ( ) SEND DATA
RECEIVE DATA 3 ( ) RECEIVE DATA
SIGNAL GROUND 7 ( ) SIGNAL GROUND
b) RS-232C Connection Without Handshake
PROGRAMMER
TTY
GROUND 1 (-)
RECEIVE DATA 3
20-mA SEND 11 (+)
20-mA RECEIVE 12 (+)
20-mA DETECT 13
c) 20-mA Current Loop Connection, Full Duplex
PROGRAMMER
TTY
GROUND 1 (-)
RECEIVE DATA 3
20-mA SEND 11 (+)
20-mA RECEIVE 12 (-)
20-mA DETECT 13 (+)
d) 20-mA Current Loop Connection, Half Duplex
NOTES:
1. All signals are named with respect to the originating unit.
2. All undesignated pins are to be left open.
3. For applications that do not require handshaking, the programmer's clear to send (CTS) line is pulled up internally.

Figure 2-3. Sample Interconnection Methods

SECTION 3
OPERATION

3.1 OVERVIEW

The LogicPak™ can be used in 29A, System 19, or 100A programmers of any configuration; see section 1.3 for hardware and firmware compatibility levels required. The LogicPak™ can obtain data from three sources: a master device, a serial port, or from the keyboard. See figure 3-1. The procedures used to perform basic operations with your LogicPak™ are described in this section. Wherever possible, key sequences have been included to use your LogicPak™ with a 29A Universal Programmer updated to Rev C firmware (read section 1.3 carefully to determine your programmer's firmware revision level). Refer to your programmer manual for key sequences for the System 19 and 100A programmers.

To program a logic device, you must first input the desired state of the device fuses into the programmer RAM. With Data I/O's Programmable Logic Development System (PLDS) you may input this fuse information in several forms: H&L programming tables for IFL devices, Boolean equations for PAL devices (PALASM), and decimal fuse number and state. The PLDS translates any one of these inputs into the LogicPak™ fuse map such as the one shown in figure 3-2; see section 1.4.1 for a detailed description of the LogicPak™ fuse map. The programmer then blows the fuses in the logic device according to the LogicPak™ fuse map previously stored in the programmer RAM. The procedure used to blow the fuse is called the programming algorithm and is stored in the P/T adapter firmware. The device is functionally tested after programming.

3.2 POWER UP

NOTE

If the LogicPak™ with adapter is not installed in the programmer before power is turned on, you will hear a beep until the LogicPak™ is installed.

When power is applied, the programmer will perform an automatic self-test routine; see section 1.4.3. When the self-test routine is complete, the programmer will signal its readiness; see your programmer manual.

To turn the programmer on:

1. Check to make sure a device is not in a socket. If a device is in a socket, lift up the lever (on the upper left of the socket; see section 3.4.2), then gently lift the device out of the socket.
2. Plug the AC power cord into the power outlet.

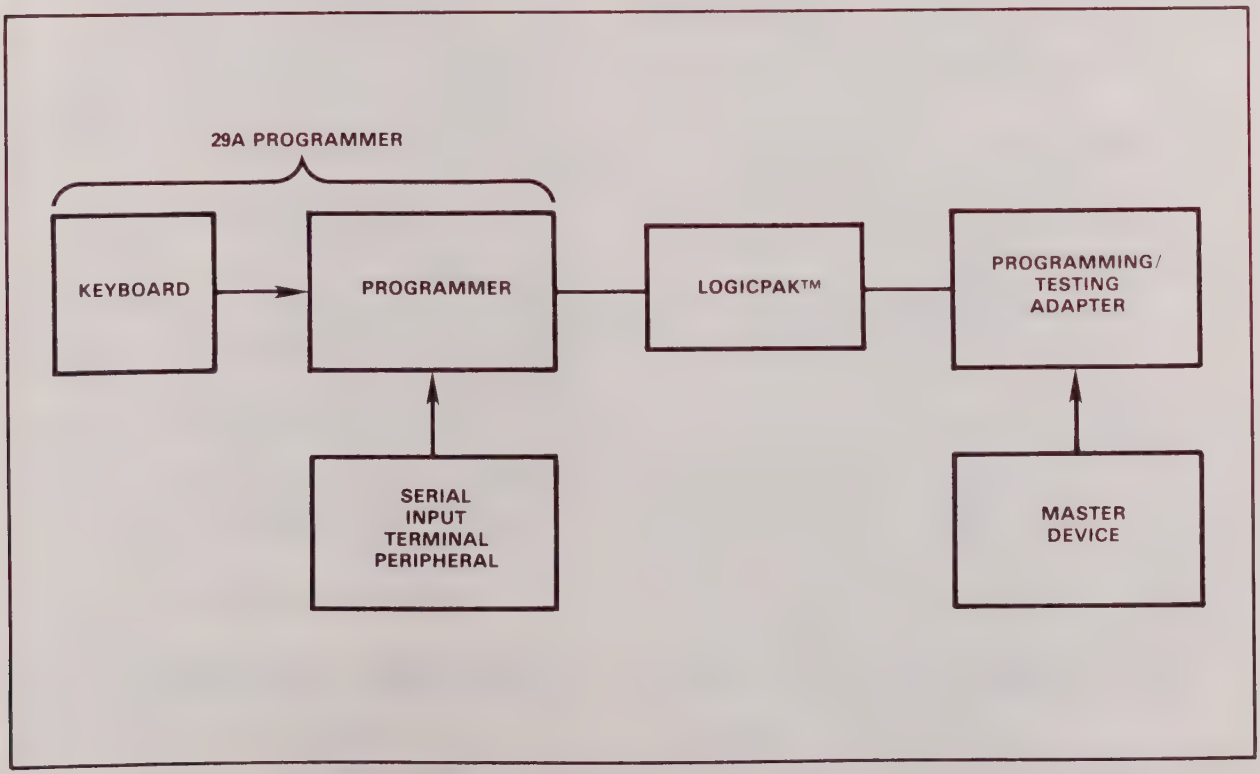


Figure 3-1. Loading Data Via the Serial Port or Master Device

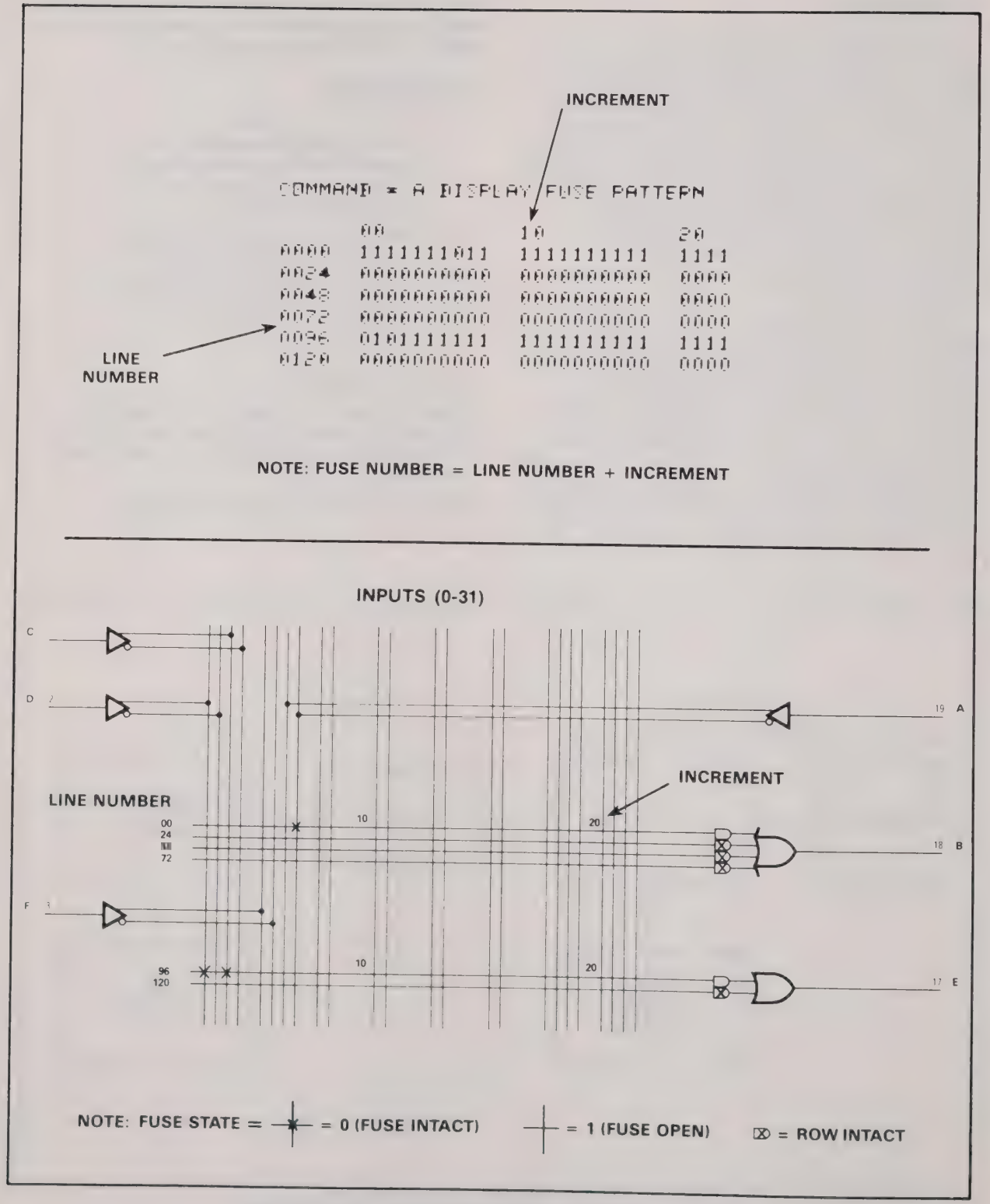


Figure 3-2. Fuse Map Applied to Logic Device Diagram

3. Flip the power switch up to the ON position; see figure 3-3.

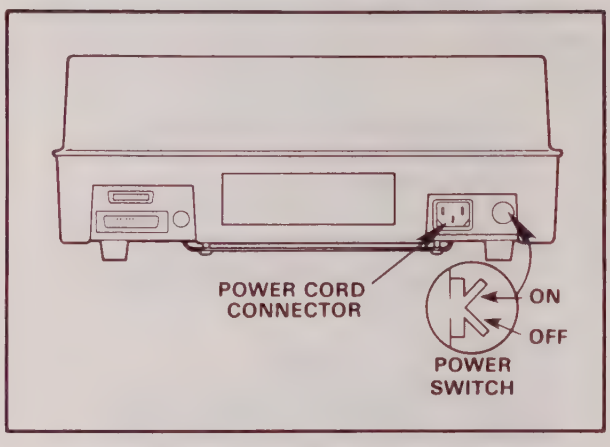


Figure 3-3. Programmer Power Switch Location

3.3 POWER DOWN

CAUTION

Do not turn the power off while the programmer is doing an operation or when a device is in a socket; voltage transients may damage the device.

To turn the programmer power off:

1. Check to make sure the programmer is not in an operation process. If it is, wait until the operation is complete.
2. Check to make sure a device is not in a socket. If a device is in a socket, remove it as described in section 3.4.3.
3. Flip the power switch down to the OFF position (figure 3-3).

3.4 BASIC DATA TRANSFER OPERATIONS

The basic operations that can be accomplished with the LogicPak™ and 29A Universal Programmer are:

- develop data (described in section 1.4.1 and design adapter manuals)
- load RAM with master device data (described in section 3.4.4)
- program a device with RAM data (described in section 3.4.5)
- verify RAM data against the device data (described in section 3.4.6)
- functionally test device (described in section 1.4.3)

The following sections describe device-related operations with the PLDS using a P/T adapter. Most setup procedures specify that you enter the family and pinout codes. Data I/O recommends that you develop the habit of entering these codes when prompted by the equipment. However, if you are using a design adapter, you will be able to perform nondevice-related operations without entering the family and pinout codes; for example, with the PALASM design adapter, Boolean equations can be entered into the PALASM editor without family and pinout codes. (Family pinout codes are automatically generated upon execution of "translate source equations" in the PALASM mode.)

If the programmer has been used to program PROMs, or for some other reason contains data in RAM, this could adversely affect the fuse pattern developed for logic devices or could inadvertently set option parameters. Therefore, execute the "clear RAM" select function (see programmer manual), or switch off the programmer (section 3.3) to clear RAM before beginning operations with the PLDS.

All data transfer or verification operations occur between the programmer's internal RAM and the device or between the RAM and serial port in your programmer. Because the operation procedure to transfer data via a serial port varies among programmers, this manual describes only data transfer using the 29A. For other programmers, refer to the specific operation manual.

NOTE

An adapter must be installed in the LogicPak™ before any of these operations can be performed (see section 2.4).

During copy and verify operations, ADDR and SIZE appear in the 29A prompts. These correspond to starting address and block size, respectively. These block limits must remain in the default state for logic device programming. An error code (see section 4, table 4-1) will be displayed if these limits are altered. For more detail on these parameters, see your programmer manual.

3.4.1 FAMILY CODE AND PINOUT CODE SELECTION

Any device that can be programmed with the LogicPak™ is specified by a unique combination of a two-digit family code and a two-digit pinout code; these codes are provided in each adapter manual. Once the codes are entered for a particular device, the LogicPak™ remains set up for any operation with that device until you enter new codes. If invalid family and pinout codes are entered, a beep will sound. In remote control operation,

will be displayed, and the operation will be stopped when a device operation is attempted.

To select the family and pinout codes:

1. Locate the manufacturer and part number stamped on the device.
2. Go to the family and pinout code table in the appropriate adapter manual and find the manufacturer's name.
3. Go to the column entitled "Device Part Number" and find the number corresponding to the number on the device.
4. Go to the columns labeled "Family Code" and "Pinout Code" to find the code numbers corresponding to the device number for the manufacturer of the device.
5. Enter the family code and pinout code you selected from this table when prompted by the programmer or terminal. An LED (light emitting diode) will light above one of the sockets on the adapter.

3.4.2 DEVICE INSERTION

Once you have entered the appropriate family and pinout codes, the LogicPak™ is ready to accept a device in the socket below the lighted LED.

A good electrical connection between the device and the socket is essential. To ensure a good connection:

1. Check to make sure the programmer is not doing an operation. If it is, wait until the operation is complete.
2. Lift the lever on the upper-left side of the socket below the lighted LED; see figure 3-4. (The lever will stay in the upright position.)
3. Gently set the device in the socket below the lighted LED. Make sure pin 1 of the device is aligned with pin 1 of the socket (upper-left corner); see figure 3-4.
4. Push the lever down to lock the device in the socket.

3.4.3 DEVICE REMOVAL

To remove a device:

1. Check to make sure the programmer is not doing an operation. If it is, wait until the operation is complete.
2. Lift the lever on the left side of the socket; see figure 3-4. The lever will remain in the upright position.
3. Lift the device out of the socket; the LED will remain illuminated.

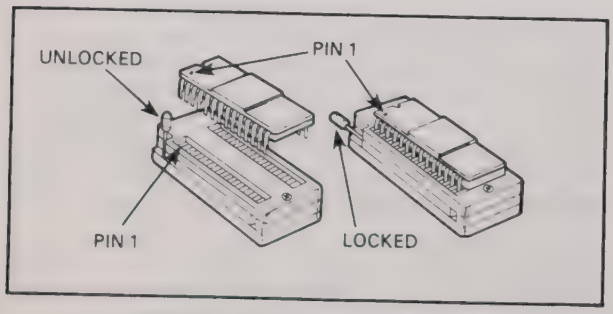


Figure 3-4. Device Installation

3.4.4 LOAD RAM WITH MASTER DEVICE DATA

To load the 29A RAM with data from a master device with control from programmer front panel, follow the steps given below.

NOTE
If options are desired (see section 3.5), select options and parameters as needed before proceeding.

1.

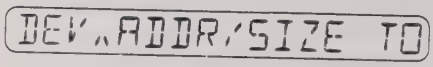
to select the mode.

29A Displays

2.

to select the source of the data.

29A Displays

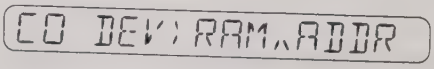


ADDR/SIZE pertains to block limit parameters. These are PROM-related and are not to be used with logic devices. Leave defaults in effect.

3.

to select the destination for the data.

29A Displays



4.

29A Displays

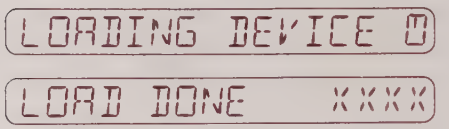
5. Enter the family code and pinout code (see section 3.4.1).

NOTE
The appropriate socket LED will light.

6. Insert the master device into the appropriate P/T adapter socket. (See section 3.4.2.)

7.

29A Displays



NOTE

XXXX is the sum-check of the device fuses.

8. Remove the master device from the LogicPak™ (see section 3.4.3)

To load the 29A with data from a master device from the terminal control mode, follow the steps given below.

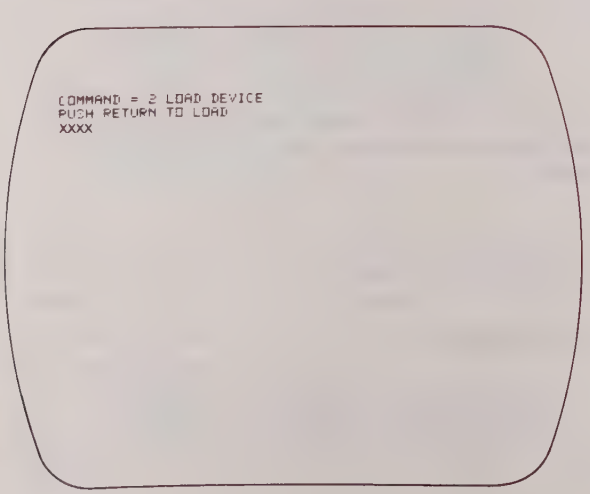
1. Place the system in terminal mode; see section 3.5.1
2. Enter the family pinout code, if prompted by the terminal.

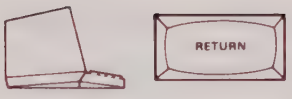
NOTE

If options are desired (see section 3.5), select options and parameters as needed before proceeding.

3.

Terminal Displays



4. 

There will be a short delay while the load operation is occurring. After the load operation is complete, the terminal will display XXXX.

3.4.5 PROGRAM DEVICE WITH RAM DATA

NOTE

If options are desired (see section 3.5), select options and parameters as needed before proceeding.

When programming a device, the system performs illegal bit tests and blank checks at nominal V_{CC}.

To program a blank device with the data in the 29A RAM with control from the programmer front panel, follow the steps given below.

1.

29A Displays

2.

29A Displays

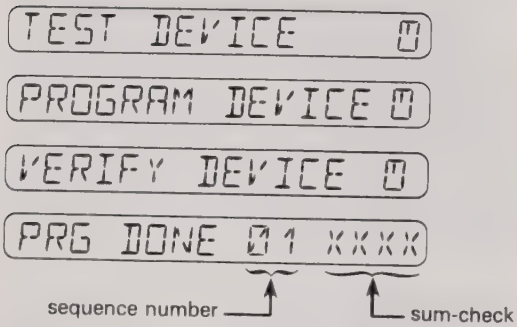
3.

29A Displays

4.

5. Enter the family code and pinout code (see section 3.4.1).
6. Insert the blank device into the adapter socket (section 3.4.2).
7.

29A Displays



(increments by 1 for each device programmed)

8. Remove the device from the LogicPak™ (see section 3.4.3).

To program a device with 29A RAM data from the terminal control mode:

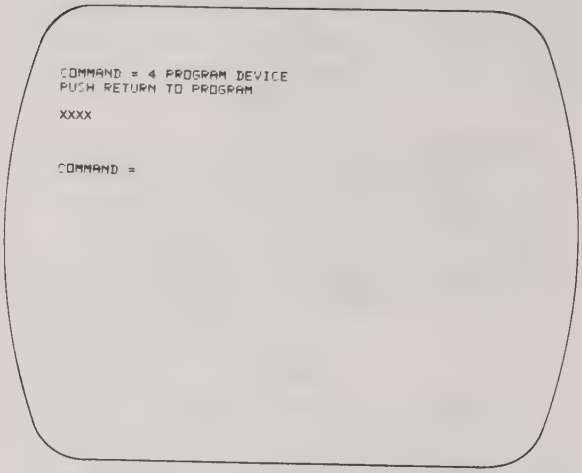
1. Place the system in the terminal mode; see section 3.5.1.
2. Enter the family pinout code, if prompted by the terminal.

NOTE

If options are desired (see section 3.5), select options and parameters as needed before proceeding.

3.

Terminal Displays



4.

There will be a short delay after pressing RETURN. This is when the programmer is pretesting, programming, verifying, and functional testing the device. If no errors occur, the terminal displays XXXX.

NOTE

XXXX is the sum-check of the device fuses.

3.4.6 VERIFY AND FUNCTIONALLY TEST DEVICE

To verify and functionally test a device from 29A front panel control follow the steps given below:

NOTE

If options are desired (see section 3.5), select options and parameters as needed before proceeding.

The verify routine compares the device data to RAM data and performs functional testing, if this option is selected.

1.

29A Displays

2.

29A Displays

3.

29A Displays

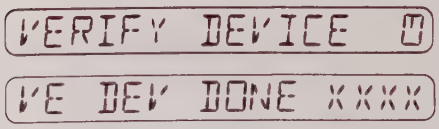
4.

29A Displays

5. Enter the family code and pinout code (see section 3.4.1).
6. Insert the device to be verified and/or tested into the LogicPak™ (see section 3.4.2).

7.

29A Displays



NOTE
XXXX is the sum-check.

8. Remove the master device from the LogicPak™ (see section 3.4.3).

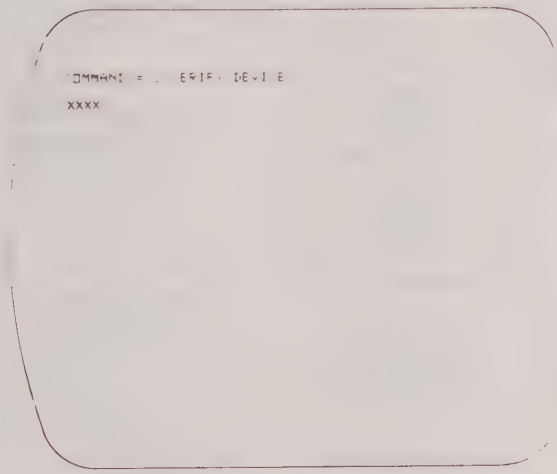
To verify and test a device from terminal control, follow the steps given below.

1. Place the system in the terminal mode; see section 3.5.1.
2. Enter the family pinout code, if prompted by the terminal.

NOTE
If options are desired (see section 3.5), select options and parameters as needed before proceeding.

3.

Terminal Displays



There will be a short delay after pressing 3 . This is when the programmer is verifying and functional testing the device. If no errors occur, the terminal displays XXXX.

NOTE
XXXX is the sum-check of the device fuses.

3.5 SYSTEM COMMANDS

In addition to the copy (load or program), verify, edit, and select functions described in the Operation Section of your programmer manual, the LogicPak™ offers numerous system commands that allow you to manipulate data and set parameters. System commands are accessed by entering a two-character select code from the programmer front panel or a one-character menu code from the terminal. Some commands will prompt for a data entry. The operational overview (figure 3-5) will help you develop data and program a device using the system commands and programmer operations. Table 3-1 lists the select codes for Data I/O programmers to enter system commands from the programmer front panel or from a terminal in terminal mode.

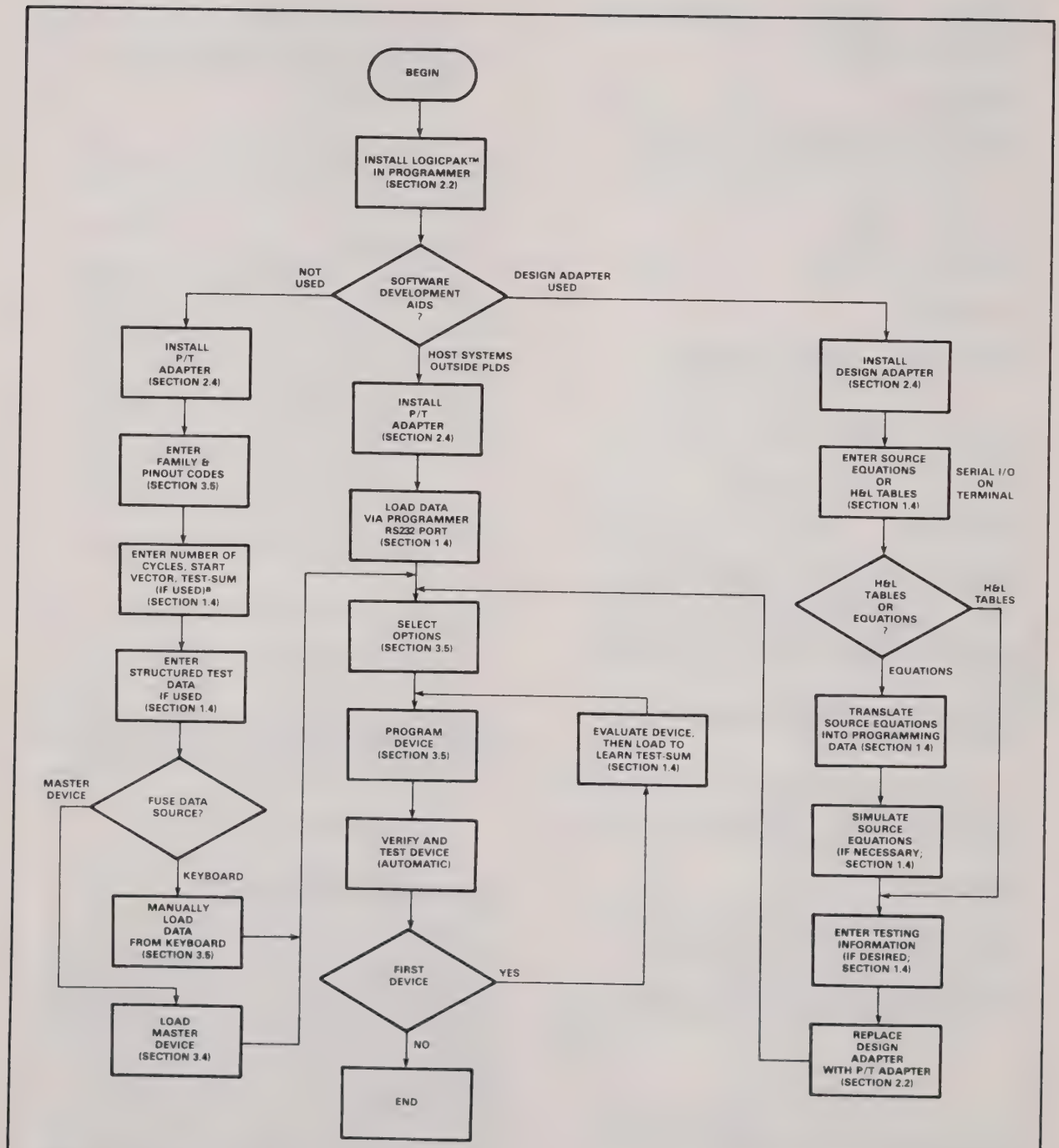

BEGIN
INSTALL LOGICPAK™ IN PROGRAMMER (SECTION 2.2)
SOFTWARE DEVELOPMENT AIDS ?
NOT USED
HOST SYSTEMS OUTSIDE PLDS
DESIGN ADAPTER USED
INSTALL P/T ADAPTER (SECTION 2.4)
ENTER FAMILY & PINOUT CODES (SECTION 3.5)
ENTER NUMBER OF CYCLES, START VECTOR, TEST-SUM (IF USED)ⓐ (SECTION 1.4)
ENTER STRUCTURED TEST DATA IF USED (SECTION 1.4)
MASTER DEVICE
FUSE DATA SOURCE?
KEYBOARD
MANUALLY LOAD DATA FROM KEYBOARD (SECTION 3.5)
LOAD MASTER DEVICE (SECTION 3.4)
INSTALL P/T ADAPTER (SECTION 2.4)
LOAD DATA VIA PROGRAMMER RS232 PORT (SECTION 1.4)
SELECT OPTIONS (SECTION 3.5)
PROGRAM DEVICE (SECTION 3.5)
EVALUATE DEVICE, THEN LOAD TO LEARN TEST-SUM (SECTION 1.4)
VERIFY AND TEST DEVICE (AUTOMATIC)
FIRST DEVICE
YES
NO
END
INSTALL DESIGN ADAPTER (SECTION 2.4)
ENTER SOURCE EQUATIONS OR H&L TABLES (SECTION 1.4)
SERIAL I/O ON TERMINAL
H&L TABLES OR EQUATIONS ?
H&L TABLES
EQUATIONS
TRANSLATE SOURCE EQUATIONS INTO PROGRAMMING DATA (SECTION 1.4)
SIMULATE SOURCE EQUATIONS (IF NECESSARY; SECTION 1.4)
ENTER TESTING INFORMATION (IF DESIRED; SECTION 1.4)
REPLACE DESIGN ADAPTER WITH P/T ADAPTER (SECTION 2.2)
ⓐ Logic Fingerprint™ test may also be derived from the master device (see section 1.4)

Figure 3-5. Operational Overview Flowchart

Table 3-1. PLDS System Command Summary

MODULE OR ADAPTER	COMMAND TYPE	FROM FRONT PANEL	VIA TERMINAL	COMMAND DESCRIPTION	SEE SECTION
LogicPak™ (with any adapter)		--	Ø	Display menu	3.5.2
		E 1	-	Enable terminal mode	3.5.1
		--	1	Enter family and pinout codes	3.5.3
		E 5	5[a]	Set reject count	3.5.4
		E 6	6	Select verify option	3.5.5
		E 7	7	Select security fuse option	3.5.6
		E 8	8	Set number of Logic Fingerprint™ test cycles	3.5.7
		E 9	8	Enter starting vector and test-sum	3.5.8
		--	8	Enter structured test vectors	3.5.9
		E A	A	Display fuse pattern	3.5.10
		E B	B	Receive fuse data	3.5.11
		E C	C	Transmit fuse data	3.5.12
		E D	D	Display sum-check of fuse data	3.5.13
		E E	E	Enter fuse data by fuse number	3.5.14
		E F	-	Display configuration number	3.5.15
		--	ESC	Before removing adapter	3.5.16
[a]Except with PALASM adapter.					
PALASM Design Adapter	Development	--	Ø	Display menu	Refer to PALASM Design Adapter Manual
		--	1	Enter family and pinout codes	
		E 2	2	Receive source equations	
		E 3	3	Transmit source equations	
		E 4	4	Translate source equations	
		E Ø	5	Simulate source equations	
		E 5	--	Set reject count	
	Edit	--	9	Edit mode (all commands listed below)	
		--		B Display line 1	
		--		C Replace text	
		--		D Delete text	
		--		E Display to end	
		--		I Insert/enter text	
		--		K Delete current line	
		--		L Display 24 lines	
		--		M Display commands	
		--		U Display previous line	
		--		(space bar) Move cursor/prompt right	
		--		(RUB/DEL) Move cursor/prompt left and delete during insert mode	
		--		CTRL H/ (back space) Move cursor/prompt left	
		--		CTRL M/ (carriage return) Display next line	
		--		CTRL P Purge all text	
		--		CTRL Z Exit editor/exit "C" or "I" modes of editor	
		--		ESC Before removing adapter	

NOTE: ESC (escape) returns control to programmer front panel.

Table 3-1. (Con't.)

MODULE OR ADAPTER	COMMAND TYPE	FROM FRONT PANEL	VIA TERMINAL	COMMAND DESCRIPTION	SEE SECTION
H&L Design Adapter	Development	--	0	Display menu	Refer to H&L Design Adapter Manual
		--	1	Enter family and pinout codes	
		E 2	2	Receive data (IFL format)[b]	
		E 3	3	Transmit data (IFL format)	
	Edit	--	4	Edit mode	
		--		G Enter gate number	
		--		P Enter product term number	
		--		T Enter transition term number	
		--		V Move cursor forward	
		--		V Move cursor backward	
		--		F Display next term	
		--		R Display last term	
		--		N Enter next field	
		--		I Insert term	
		--		D Delete term	
		--		C Clear term	
		--		X Deactivate term	
		--		E Display edit sub-menu	
		--		(0) Exit edit mode	
		--		(1) Return to edit mode	
		--		(2) Serial input (receive IFL format)[b]	
		--		(3) Serial output (transmit IFL format)	
		--		(4) List low-order terms	
		--		(5) List high-order terms	
		--		(6) Compress terms	
		--		CTRL Z Exit edit mode	
		--		ESC Before removing adapter	
[b]Integrated fuse logic, Signetics ASCII					
All P/T Adapters	Device	--	1	Enter family and pinout codes	3.4.1
		Load	2	Load fuse data from device to RAM	3.4.3
		Verify	3	Verify fuse data and perform functional test	3.4.5
		Program	4	Program device with RAM data	3.4.4

NOTE: ESC (escape) returns control to programmer front panel.

NOTE

The sequence explanations assume no operating errors. If these occur, the programmer signals audibly (except in remote control) and displays a two-digit error code. It also beeps once when an incorrect key is pressed. Error codes are explained in section 4.1 (table 4-1) and in your programmer manual. Some errors will return you to the programmer front panel control.

3.5.1 ENABLE TERMINAL MODE

Select code E1 transfers control of the PLDS to the terminal. After control is transferred, the 29A will display only its action symbol. This command allows you to access data development and remote operations resident in the design adapters and remote operations using the P/T adapters.

The terminal will prompt you to enter a family pinout code unless one has been entered. The terminal will display:

```
                                        COPYRIGHT 1982
DATA I/O CORPORATION    PLDS-PROGRAMMING TESTING ADAPTER

 0 - DISPLAY MENU
 1 - ENTER FAMILY PIN CODE
 2 - LOAD DEVICE
 3 - VERIFY DEVICE
 4 - PROGRAM DEVICE
 5 - ENTER REJECT COUNT OPTION
 6 - ENTER VERIFY OPTION
 7 - ENTER LAST FUSE OPTION
 8 - ENTER FUNCTIONAL TEST DATA
 A - DISPLAY FUSE PATTERN
 B - RECEIVE FUSE DATA
 C - TRANSMIT FUSE DATA
 D - DISPLAY FUSE SUMCHECK
 E - ENTER DECIMAL FUSE DATA

 ESC> - BEFORE REMOVING ADAPTER

COMMAND = _
```

Enter a family pinout code; see section 3.4.1. Once it has been entered, the terminal will display the command menu; see section 3.5.2.

See section 2.6 for terminal setup procedure.

3.5.2 DISPLAY COMMAND MENU

This command causes the PLDS to display its command menu on the terminal, as shown below:

```
COMMAND = 0 DISPLAY MENU                COPYRIGHT 1982

DATA I/O CORPORATION    PLDS-PALASM DESIGN ADAPTER

 0 - DISPLAY MENU
 1 - ENTER FAMILY PIN CODE
 2 - RECEIVE SOURCE EQUATIONS
 3 - TRANSMIT SOURCE EQUATIONS
 4 - TRANSLATE SOURCE EQUATIONS
 5 - SIMULATE SOURCE EQUATIONS
 6 - ENTER VERIFY OPTION
 7 - ENTER LAST FUSE OPTION
 8 - ENTER FUNCTIONAL TEST DATA
 9 - EDIT SOURCE EQUATIONS
 A - DISPLAY FUSE PATTERN
 B - RECEIVE FUSE DATA
 C - TRANSMIT FUSE DATA
 D - DISPLAY FUSE SUMCHECK
 E - ENTER DECIMAL FUSE DATA

 ESC> - BEFORE REMOVING ADAPTER

COMMAND =
```

3.5.3 FAMILY CODE AND PINOUT CODE

From the 29A front panel control, family pinout code entry is part of a device-related operations; see sections 3.4.1 and 3.4.4.

Terminal Displays

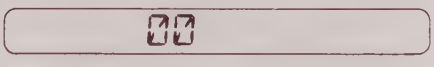
```
COMMAND = 1 ENTER FAMILY PIN CODE
FAMILY PIN CODE (0000
```

Enter the family pinout code. See section 3.4.1 for more detail.

3.5.4 SET REJECT COUNT OPTION

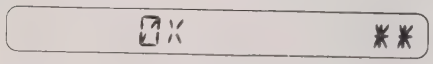
This command allows you to select the number of programming pulses applied to the device fuses before the programmer rejects the device as unprogrammable. The default value of 00 selects the manufacturer's specified number of programming pulses. Refer to the adapter manual for specific entries to select optional reject values for single-pulse, military, etc., programming specifications.

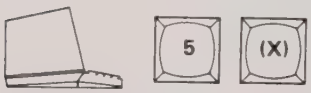
29A Displays



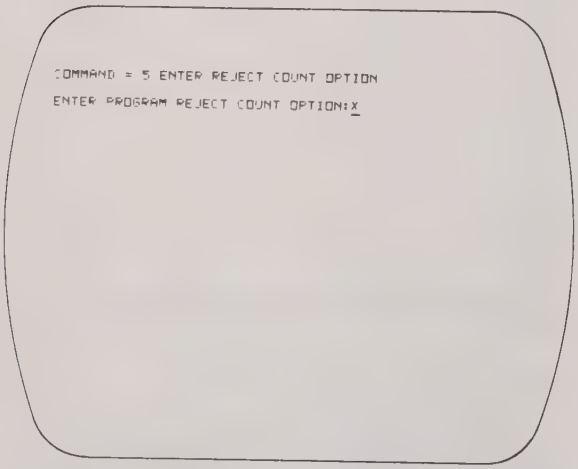
To change the reject count to an optional value, enter the code number (X) specified in the adapter manual.

29A Displays





Terminal Displays



NOTE
PALASM adapters do not use this option.

3.5.5 SELECT VERIFY OPTION

Three options are available for selecting verify and functional test routines. These routines are described in detail in secitons 3.4.6 and 1.4.3.

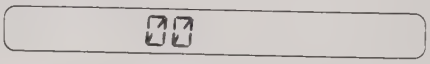
Options available are:

OPTIONS	DESCRIPTION
	Default option. Perform fuse verify, followed by structured test (if test vectors are present in RAM), and Logic Fingerprint™ test (if one or more Logic Fingerprint™ test cycles are selected), in that order.
	Perform fuse verify only.
	Perform structured test and Logic Fingerprint™ test only, in that order. Do not perform fuse verify.

Option 0 (default) is the option used in normal operation. Option 1 checks the programming of the device fuses without checking its functionality. Use option 2 to functionally test devices with the security fuse blown. In addition, option 2 can be used to learn the Logic Fingerprint™ test of a device with the security fuses blown. Fuse data in RAM will be cleared during this operation. Programming cannot occur with option 2 selected.

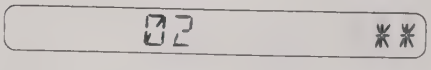
Test options must be entered from the programmer's keyboard or a terminal. The option will remain in effect until it is changed or until the unit is powered down. To reselect the default, key in option 0.

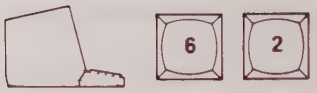
29A Displays



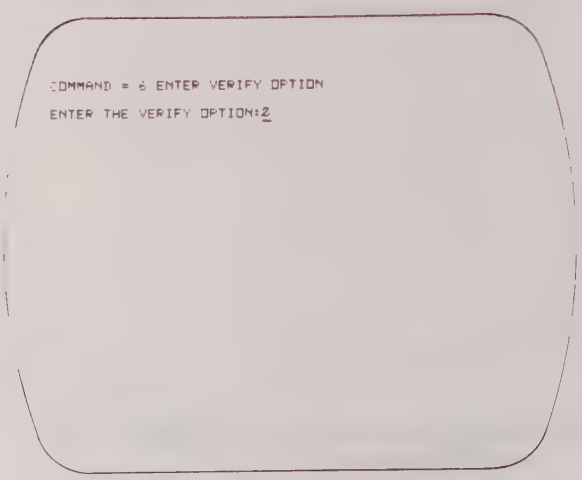
For example, to select functional test **only**:

29A Displays





Terminal Displays



3.5.6 SELECT SECURITY FUSE OPTION

Some logic devices are equipped with protective fuses called security fuses. Once the security fuses are programmed, the fuse states in the logic array cannot be copied. Programming the security fuses makes it very difficult to pirate a device design.

The PLDS security fuse programming feature is a fail-safe function. You can either enable programming of the security fuse at all times, only allow programming when security fuse data are downloaded to the PLDS via the serial port, or disable programming completely, whether security fuse data are downloaded or not.

When the security fuse has been blown, a Logic Fingerprint™ test and structured test can still be performed, but a fuse verify operation is not possible. See section 3.5.5.

To enable programming of security fuses two conditions must be met: 1) the security fuse state in the programmer RAM must be 1 (or true), and 2) security fuse programming must be enabled. Once the security fuse option is selected, it will remain in effect until changed or until the programmer is turned off.

When security fuse data are entered into RAM in the JEDEC ASCII-logic format (see section 1.4.1), data in the **G** field indicate the state of the security fuse. The **G** field does not affect the enable state of the security fuse option; the enable state must be entered separately. This can be done before or after loading JEDEC ASCII-logic format data.

Security fuse states cannot be loaded from a master device.

CAUTION

Once the security fuse is programmed, you can no longer verify the state of any fuse in the device. The process cannot be reversed; therefore, be certain that you want to program the security fuse before you activate this function.

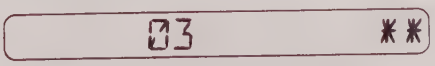
Security fuse select-code options are:

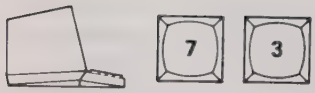
OPTION	DESCRIPTION
0	Default option. Disable programming and set the security fuse state in RAM to 0.
1	Disable programming, and set security fuse state in RAM to 1 (true).
2	Enable programming, and set security fuse state in RAM to 0. (Data downloaded in the JEDEC format can change the security fuse state to 1.)
3	Enable programming, and set security fuse state in RAM to 1. (Data downloaded in the JEDEC format can change the security fuse bit back to 0.)

29A Displays

For example, to enable security fuse programming and set security fuse state in RAM to 1 (option 3):

29A Displays





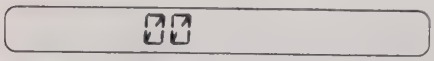
Terminal Displays

```
COMMAND = 7 ENTER LAST FUSE OPTION
ENTER LAST FUSE OPTION:3
```

3.5.7 SET NUMBER OF LOGIC FINGERPRINT TEST CYCLES

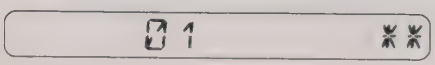
This command allows you to select the number of test cycles that are performed during the Logic Fingerprint™ test, which is described in section 1.4.3. The default value is **00**, which disables the Logic Fingerprint™ test.

29A Displays



For example, to enable one cycle of testing,

29A Displays



and enter the number of cycles desired.

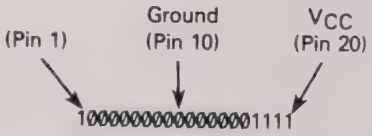
Terminal Displays

```
COMMAND = 3 ENTER FUNCTIONAL TEST DATA
CYCLES FOR FINGERPRINT: 01
```

3.5.8 ENTER LOGIC FINGERPRINT™ TEST STARTING VECTOR

This command enables you to view or enter the starting test vector and resulting test-sum for the Logic Fingerprint™ test (see section 1.4.3 for a full description of this test).

For this example we will use an arbitrary starting vector for a 20-pin device of:



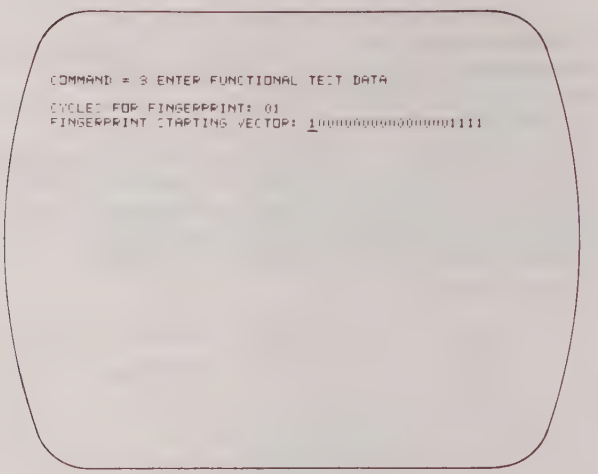
NOTE

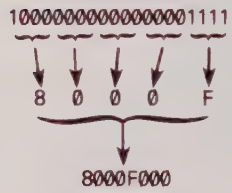
A "1" represents a high; a "0" represents a low to be applied to a particular device pin.

Values entered for ground and V_{CC} are included and affect the test-sum, but have no effect on the device under test.

Terminal mode allows the starting vector to be entered bit-by-bit after entering the number of cycles for the Logic Fingerprint™ test. For programmer front panel operation, the vector must be represented by hexadecimal numbers as shown on the preceding page.

Terminal Displays





Starting vector (binary)

(hexadecimal)

Starting vector (hexadecimal)

The unused portion of the 32-bit vector is assumed to be zeroes and must be included in the hexadecimal vector entry.

For example:

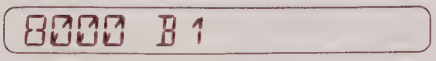
29A Displays

NOTE

The eight-character starting vector is entered into the programmer in two fields. B1 identifies the first field.

To enter the first four hexadecimal digits,

29A Displays



29A Displays

B2 represents the second field.

Enter the remaining hexadecimal digit by pressing

29A Displays

The zeroes are ignored, but are needed to correctly position the "F."

Assume that this vector, when applied to a logic device, gives the following test-sum in hexadecimal:

8F66FDAF

The E1 field represents the first four characters of the test-sum. These can be viewed or entered at this time.

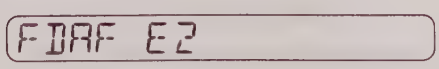
Enter the first four characters of the test-sum.

29A Displays

To view or enter the next four characters (E2 field)

Enter the next four characters.

29A Displays



Terminal Displays

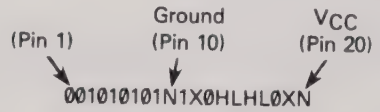
```
COMMAND = 8 ENTER FUNCTIONAL TEST DATA

CYCLES FOR FINGERPRINT: 01
FINGERPRINT STARTING VECTOR: 1000000000000000001111
FINGERPRINT: 8F66FDAF
```

3.5.9 ENTER STRUCTURED TEST VECTORS

After the Logic Fingerprint™ test parameters are entered, the terminal entry of structured test vectors can begin. Structured test vectors cannot be entered from the programmer front panel. (See section 1.4.3 and appendix A for a detailed explanation of structured tests.)

This example shows entry of the following structured test vector from a terminal:




```
COMMAND = 8 ENTER FUNCTIONAL TEST DATA

CYCLES FOR FINGERPRINT: 01
FINGERPRINT STARTING VECTOR: 1000000000000000001111
FINGERPRINT: 8F66FDAF
STRUCTURED TEST VECTOR 0001: 00101010 1N1X0HLHL0XN
```

3.5.10 DISPLAY FUSE PATTERN

This command outputs the fuse pattern in the programmer data RAM to the serial port of the programmer. The fuse states are a series of 1s and 0s representing whether a fuse is blown (1) or left intact (0). Each fuse can be identified by a decimal number (figure 3-6). The fuse states are arranged in a matrix that corresponds to the logic diagram of the device. This is useful for transferring a fuse pattern to a logic diagram for documenting a device fuse pattern. See figures 3-6 and 3-7. Logic diagrams for all devices, complete with fuse numbers, are in the P/T adapter manuals.

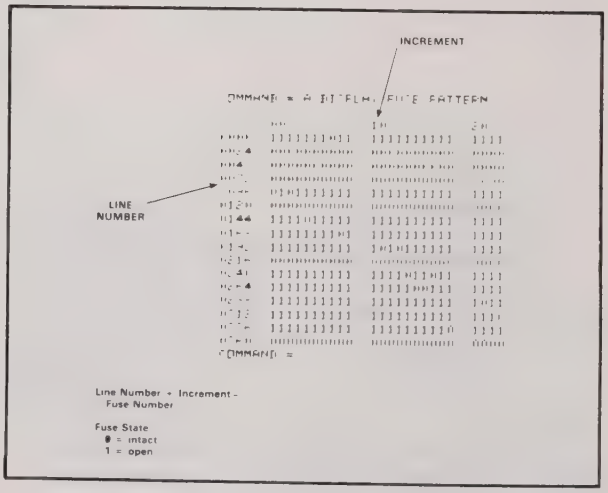
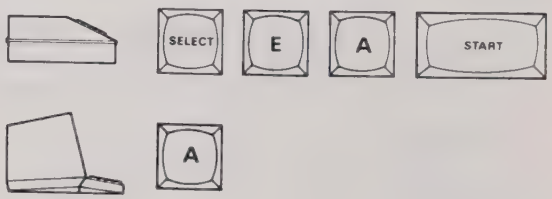
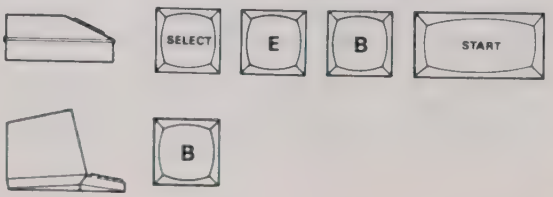


Figure 3-6. Complete Fuse Pattern



3.5.11 RECEIVE FUSE DATA

This command prepares the programmer to receive fuse data from a peripheral via the programmer serial port in the JEDEC standard ASCII-logic format (appendix A).



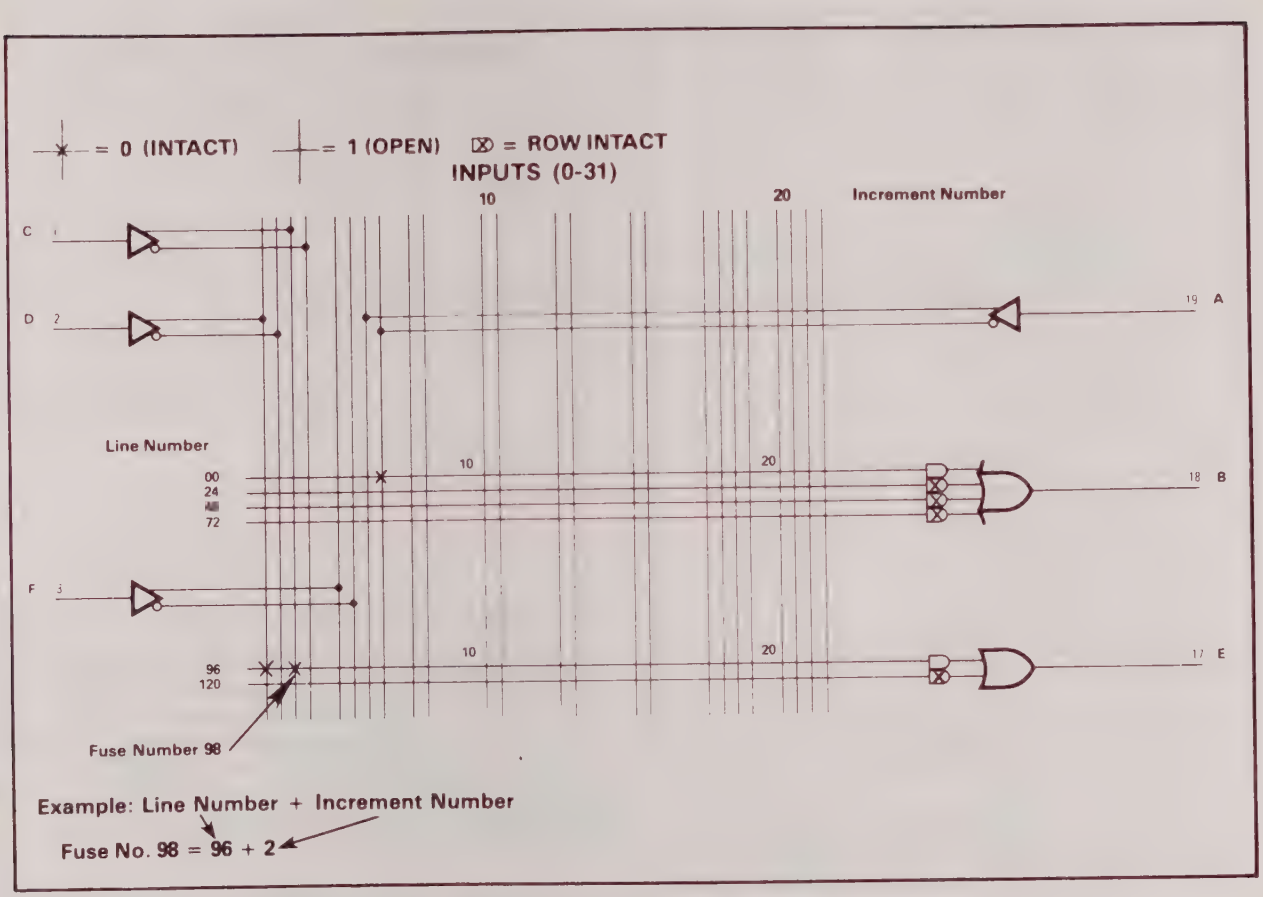
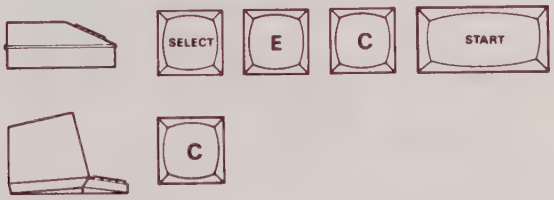


Figure 3-7. Complete Logic Diagram

Terminal Displays

3.5.12 TRANSMIT FUSE DATA

This command prepares the programmer to output fuse data via the serial port in the JEDEC ASCII-logic format (appendix A). See figure 3-8.



3.5.13 DISPLAY SUM-CHECK OF FUSE DATA

This command displays the sum-check of the fuse states in the programmer data RAM (the C field in the JEDEC format). The sum-check should match one from a previously programmed device that is known-good.

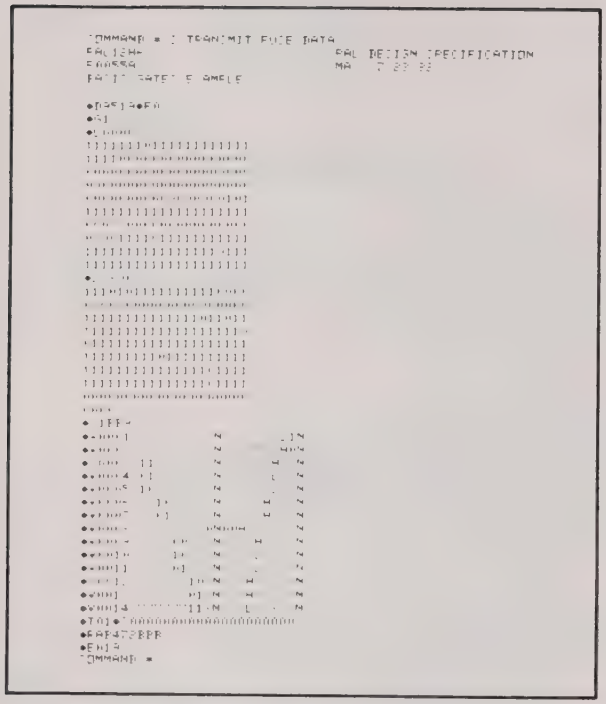
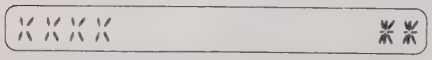


Figure 3-8. Sample Printout of JEDEC Format

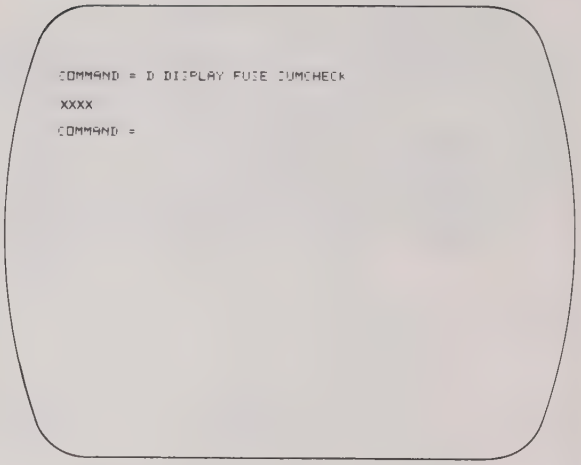
29A Displays



NOTE

XXXX is the fuse array sum-check of the device.

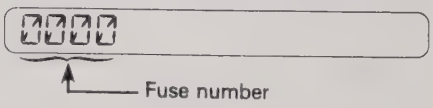
Terminal Displays



3.5.14 ENTER FUSE NUMBER AND STATE

Programming information in RAM for the logic device fuses may be entered using this command. Refer to figure 3-2 for an example of fuse states by fuse numbers. Each P/T adapter manual contains logic diagrams, complete with fuse numbers, for the devices it supports. A fuse number is obtained by adding the line number and increment number corresponding to the intersection where the fuse is located. (See figure 3-9.)

29A Displays



Enter a decimal fuse number; for example, 98.

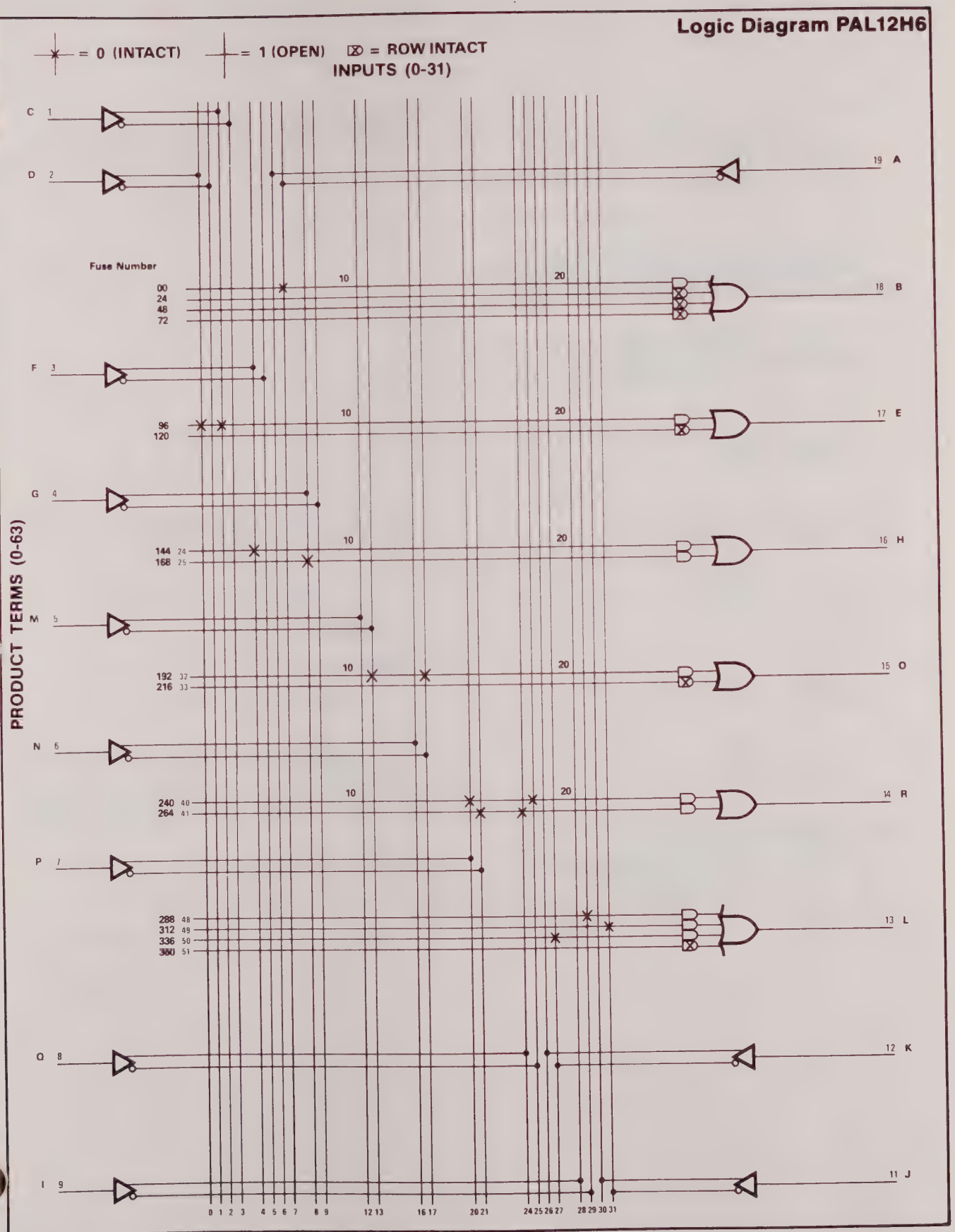
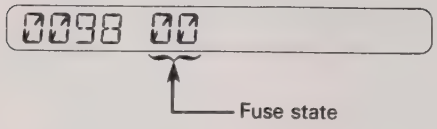


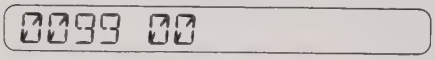
Figure 3-9. Logic Diagram

29A Displays



This display indicates that RAM data for fuse 98 is set for "don't program." To change it to a program state:

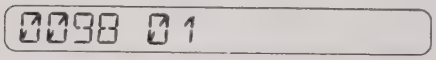
29A Displays



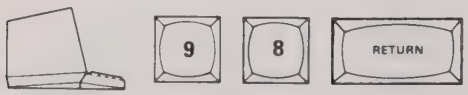
(Fuse number increments automatically.)

To decrement a fuse number:

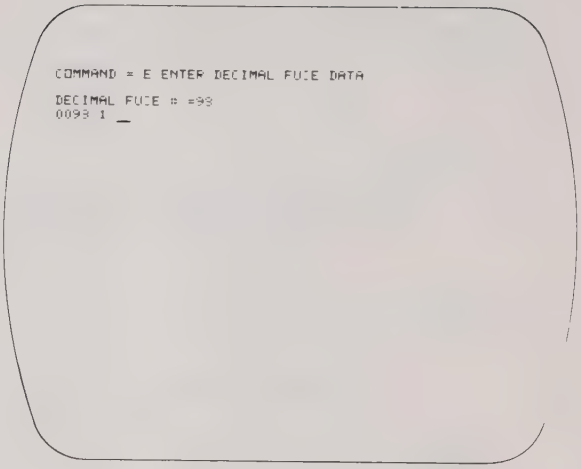
29A Displays



Enter the fuse number; for example, 98.

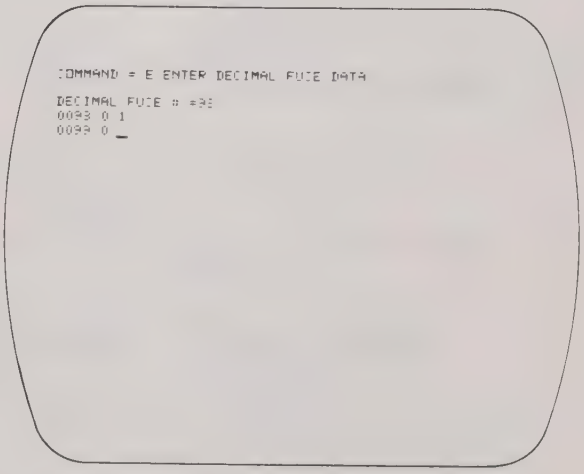


Terminal Displays



This display indicates that RAM data for fuse 98 is set for "don't program." To change to a program state:

Terminal Displays



This indicates that fuse 98 will program, increments the fuse number display, and indicates the state of fuse 99 in RAM. To display the next fuse,

To display the previous fuse,

To jump to a new fuse location,

then the decimal fuse number

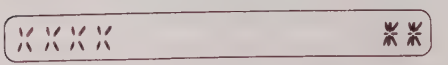
or

simultaneously to exit the edit mode.

3.5.15 DISPLAY CONFIGURATION NUMBER

This command displays the configuration number of the adapter firmware. Configuration numbers are used as serial numbers for firmware.

29A Displays



NOTE

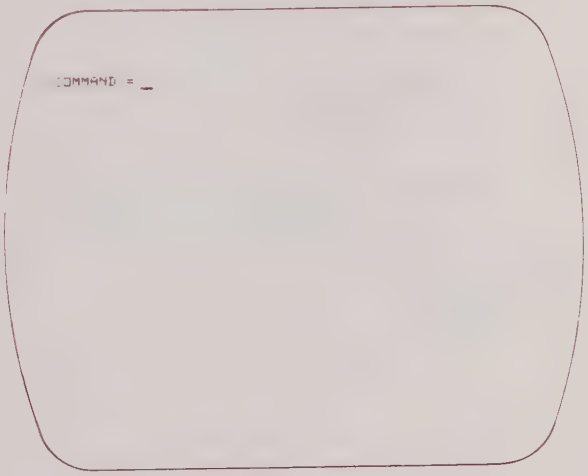
XXXX is the configuration number of the firmware in the adapter plugged into the PLDS.

3.5.16 EXIT COMMAND

During terminal mode, use this function to exit specific operating modes. When using the design adapters, this function is also used to terminate the change, insert, and edit modes.

simultaneously

Terminal Displays



returns control to the programmer front panel from terminal control.

SECTION 4
MAINTENANCE/CALIBRATION/ TROUBLESHOOTING

4.1 OVERVIEW

The support material in this section has been provided to help you keep your LogicPak™ in optimum operating condition. General maintenance practices are discussed in section 4.2, and the basic troubleshooting setup is described in section 4.4. For those LogicPak™ users who prefer to do their own calibration, detailed procedures, including measurement charts and timing diagrams for each device, are provided in each adapter manual. The basic procedures to set up the LogicPak™ in the calibration mode are described in section 4.3 of this manual.

4.2 MAINTENANCE

Regular maintenance of the LogicPak™ consists of cleaning (section 4.2.1) and inspection (section 4.2.2).

4.2.1 Cleaning

Inspect the LogicPak™ inside and out for accumulated dirt or dust. To clean the LogicPak™:

1. Wipe any dust or dirt off the outside of the LogicPak™ with a clean, damp cloth.

NOTE

Do not use abrasive cleaners or solvents that will etch the paint.

2. Remove dust from the circuit boards with a blast of dry, compressed air or a clean, soft-bristled brush.

4.2.2 Inspection

You can help prevent malfunctions by periodically inspecting your LogicPak™. Check cable connections, card seating, mounting of socketed components, etc., for shorts, opens, or unstable continuity.

If you find heat-damaged components, be particularly careful to find and correct the cause of the overheating. This will prevent further damage.

4.3 CALIBRATION

The need for calibration varies with the amount of use your LogicPak™ receives. Generally, we suggest calibration whenever: 1) programming yields fall below the manufacturer's recommended minimums, 2) when troubleshooting has been completed, or 3) if the user's company policy requires periodic calibration certification.

Because the LogicPak™ must be calibrated with an adapter installed and the values vary with different adapters, the detailed calibration procedures, measurement charts, and timing diagrams are provided in each adapter manual. The calibration setup procedure is described in this section.

NOTE

If calibration or repair is required, but you lack the facilities to accomplish it, contact the nearest Data I/O Service Center. Because of the different programmer mainframes and adapter modules, this manual does not attempt to cover all areas of programmer calibration. Instead, it lists the steps necessary to calibrate only the LogicPak™.

The first step in testing the LogicPak™ hardware is to measure the programmer power supplies (and adjust if necessary). Proper LogicPak™ operation requires calibrated programmer power supplies. These measurements are essential before any tests are performed on the LogicPak™ itself. Refer to your programmer manual for power supply adjustment.

To prepare the LogicPak™ for calibration:

1. Remove the adapter (if any) from the LogicPak™.
2. Remove the four phillips-head screws from the LogicPak™ cover (see figure 4-1).
3. Remove the two allen screws on each side of the LogicPak™ cover (see figure 4-1).

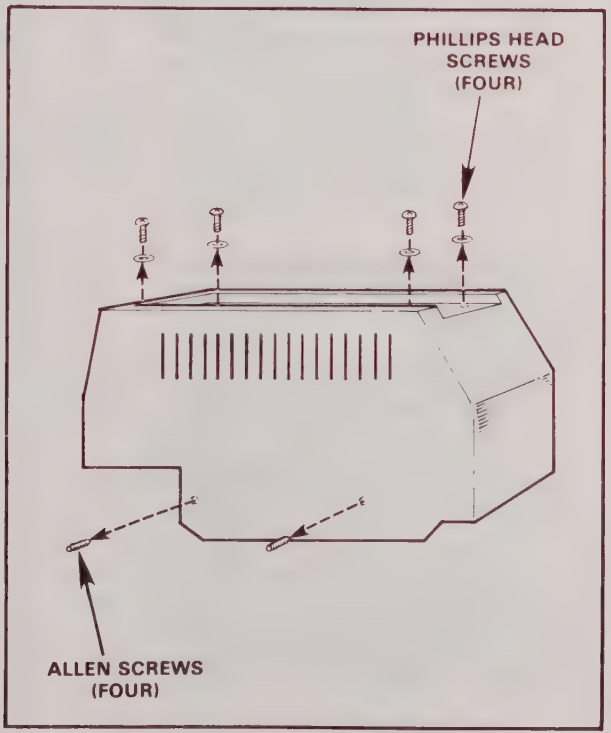


Figure 4-1. LogicPak™ Cover Removal

4. Lift the cover off the circuit board cage assembly.
5. Plug the adapter into the connector on the pin driver board as shown in figure 4-2.

To calibrate the LogicPak™ follow the series of steps presented in the measurement chart located in each adapter manual. The second step on the measurement chart is a "go/no-go" functional test of the module.

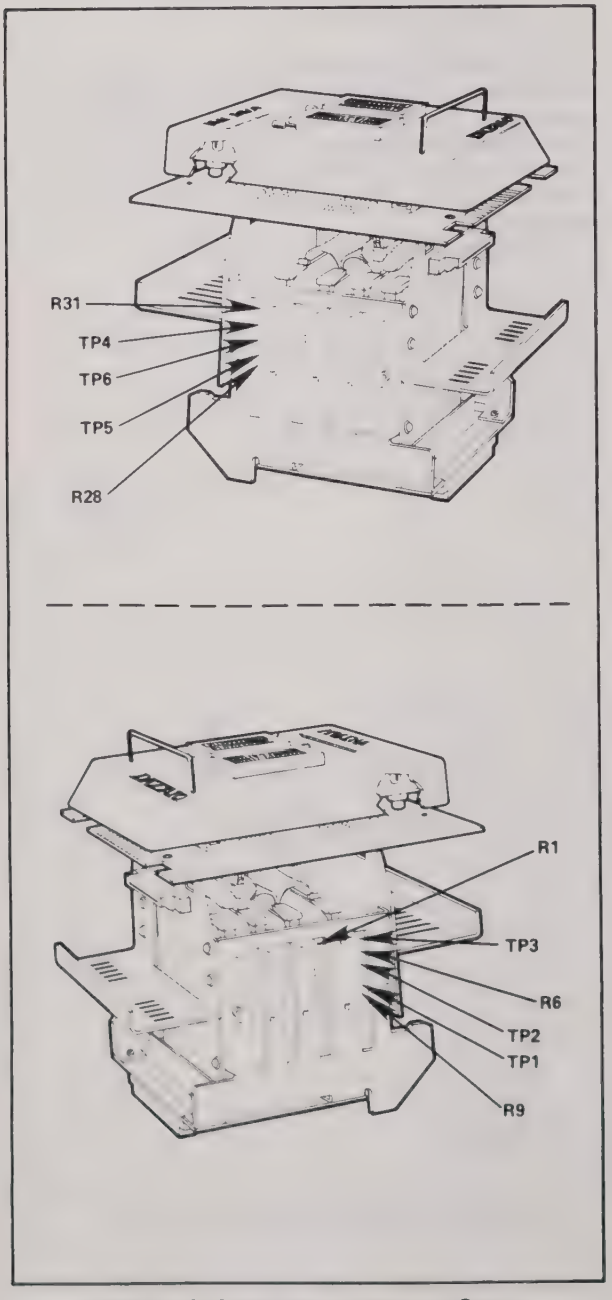


Figure 4-2. Calibration Equipment Setup

CAUTION

If the LogicPak™ fails the second step on the measurement chart, DO NOT proceed to the next step. The hardware must pass this step or further testing may damage the LogicPak™.

Steps 1 through 4 and step 16 serve to calibrate the module. Potentiometers adjust each digital-to-analog converter (DAC) to the optimum voltage. Measurements are made at test points on the LogicPak™, shown in figure 4-3 or at socket pins on the P/T adapter.

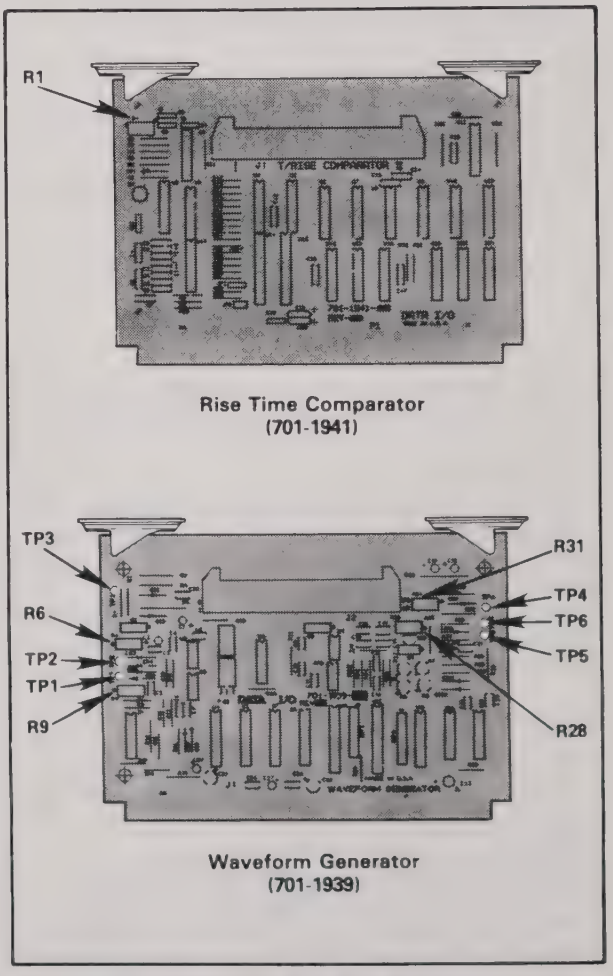


Figure 4-3. LogicPak™ Test and Adjustment Locations

If the LogicPak™ fails any step on the measurement chart, do not continue to the next step. Refer to table 4-1, which lists suspect boards and components that would cause the module to fail for each step. Subsequent tests will not give valid results unless all preceding steps are passed and adjustments made.

Table 4-1. LogicPak™ Error Codes

ERROR CODE	DESCRIPTION	ACTION
25	No Socket Adapter	Insert appropriate socket adapter.
30	No (or Invalid) Device Selected	Enter valid device family and pinout codes (refer to Comparison Chart of Programmable Logic Device in each adapter manual).
31	Overcurrent	Hardware error in LogicPak™ or shorted device. Substitute a known-good device or consult the troubleshooting section. If error 31 is displayed, caused by V_{CC} overcurrent, code 32 will display because 32 is tested before 31.
32	Backward Device	(1) Device plugged in backward; turn it around. (2) See error 31.
33	Source Buffer Write Error (RAM)	Source equations exceed the available RAM space; therefore, (1) Reduce the equation length to fit available RAM. (2) Add more RAM to system to accommodate the equation length (refer to programmer manual to expand RAM).
35	Source Equation Translation Error	Check equation buffer by connecting terminal to examine the equation buffer. This error code lets the operator know that an error exists in the source equations when the programmer is not controlled by a terminal.
36	Begin RAM Pointer Not = 0000	Refer to programmer manual to reset the begin RAM pointer to zero. This error usually occurs when changing from one programming pak to another.
37	Invalid Device-Related Operation	Verify, program, or other illegal operation was attempted, with a design adapter installed.
38	Calibration Step Error	Indicates that you've selected an incorrect calibration step. The error will also occur if a program operation is attempted prior to exist calibration. (1) Exit the calibration mode (refer to the programmer manual). (2) Reenter the correct calibration step number.
63	RAM Write Error	System RAM failure. Refer to programmer manual or contact Data I/O service representative.
65	Firmware Sum-Check Error	Contact Data I/O service representative. This indicates that the EPROM firmware in the LogicPak™ or adapter may have changed since the unit was shipped. Do not continue operation until the situation is corrected.
70	DAC Error V_{CC}	See section 4.4 (troubleshooting).
71	DAC Error Bit Switch Number 1	See section 4.4.

Table 4-1. Continued

ERROR CODE	DESCRIPTION	ACTION
72	DAC Error Bit Switch Number 2	See section 4.4.
73	DAC Error CE	See section 4.4.
74	Logic Fingerprint™ Test Verify Error	Indicates a Logic Fingerprint™ error. (1) Device passed fuse verify but failed Logic Fingerprint™ Test—defective device. (2) Operator has entered wrong test-sum. (3) Device cannot be tested with Logic Fingerprint™ (refer to p/t adapter manual for the limitations of the Logic Fingerprint™ test).
75	Structured Test Verify Error	(1) The device passed fuse verify but failed structured test—defective device. (2) Check structured test vectors and make sure they are correct. If not, reenter the correct vectors. The vector could be invalid, or the operator may have miskeyed a valid vector.
76	Self-Test Error	Indicates failure in the LogicPak™. Consult section 4.4 (troubleshooting) or contact your Data I/O service representative.
77	Security Fuse Programming Error	(1) Indicates that the security fuse option cannot be programmed in the installed device. (2) There is no security fuse option available for this type of device.
78	No Fuse Verify Set	Indicates you've tried to program the device with the verify-option mode set for 2. The verify option won't allow this. When this error code displays, select E6 and enter Ø or 1, and then you will be allowed 1 program.
82	Checksum Error	Indicates an incorrect transmission data from a peripheral to the serial port, including fuse data, CRs, STX, etc.
84	Sum-Check Error	(1) Indicates an error in the fuse data, or (2) Received fuse data does not match fuse sum-check (C-field error).
91	Fuse Address Error	Indicates an invalid fuse address. Check input and make sure fuse address is four decimal numbers or an otherwise valid address.

4.4 TROUBLESHOOTING

This section will help you interpret and isolate failures in the LogicPak™. Use it in conjunction with the calibration section in each adapter manual and section 5 (Circuit Description) and the schematics provided in the back of this manual.

There are three major classes of failures that can occur in a system composed of a programmer and a LogicPak™. The first is no system operation, the second is poor yields, and the third is LogicPak™ failure.

After successfully troubleshooting the LogicPak™, it must be calibrated according to the instructions in each adapter manual. It is very important that you calibrate the programmer before you calibrate the LogicPak™. Refer to your programmer manual for detailed programmer calibration procedures.

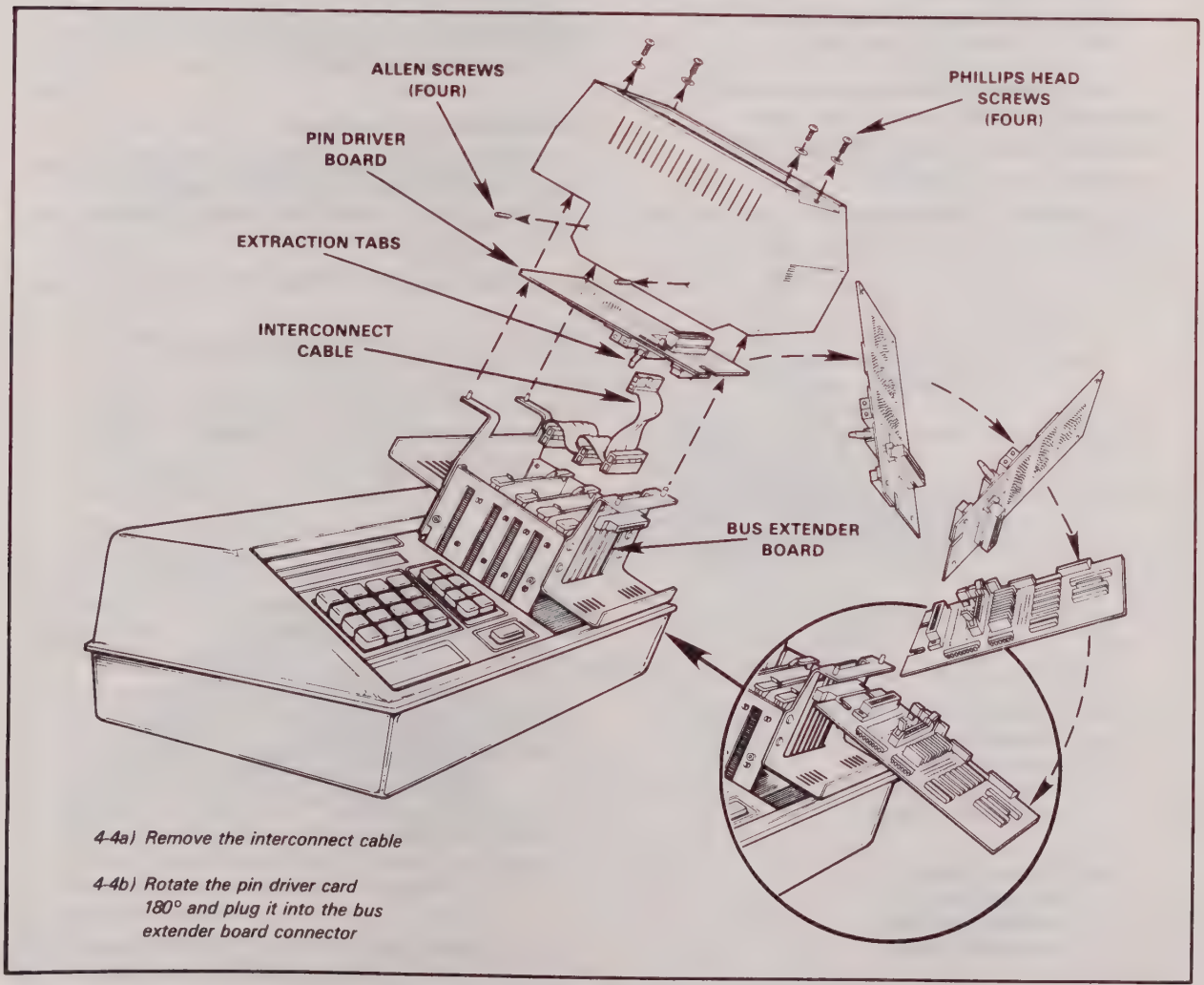
4.4.1 Equipment Set Up

The following equipment is necessary to troubleshoot the LogicPak™:

- Data I/O extender card (part number 701-1949) and service cable (part number 709-0078)
- one phillips screwdriver
- one 1/16-inch allen wrench

To prepare your LogicPak™ for troubleshooting:

1. Turn off the programmer power (section 3.3).
2. Remove the LogicPak™ from the programmer (section 2.3).
3. Remove the adapter (if any) from the LogicPak™.
4. Remove the four phillips-head screws from the LogicPak™ cover (see figure 4-4).



4-4a) Remove the interconnect cable

4-4b) Rotate the pin driver card 180° and plug it into the bus extender board connector

Figure 4-4. LogicPak™ Disassembly

5. Remove the two allen screws on each side of the LogicPak™ cover (see figure 4-4).
6. Lift the cover off the card cage assembly.
7. Unplug the pin driver board (702-1943-001) from the bus extender board and lift it slightly until you can see the interconnect cable and its connector (figure 4-4).
8. Flip the extraction tabs out on each side of the connector (figure 4-4).
9. Pull the cable out of the connector.
10. Rotate the pin driver board 180° and plug it into the connector on the front of the bus extender board (figure 4-5).

NOTE

The interconnect cable must be replaced by the service cable before you can troubleshoot your LogicPak™.

11. Lift the extraction tabs on each side of the three forward circuit cards as shown in figure 4-4.
12. Unplug the interconnect cable from the three connectors shown in figure 4-4.
13. Lift the extraction tabs on the waveform generator card (figure 4-5).
14. Pull out the waveform generator board along the guides (figure 4-5).
15. Gently push the service extender card (701-1949) along the guides to plug it into the J2 connector on the motherboard (702-1938) where the waveform generator was located.

NOTE

Any board can now be placed on the extender board. The waveform generator card (701-1939) must be either on the extender board or in the slot just behind it.

16. Firmly, but gently, push the service cable into the connectors on the three boards (as shown in figure 4-5). Notice that the extraction tabs will move back to their locked positions when the cable is locked into the connector.
17. Gently push the extender cable into the connector on the pin driver board (see figure 4-5).

4.4.2 No System Operation

The following steps should be performed if the system will not initialize with the LogicPak™ installed. After completing each step, determine whether the problem still exists.

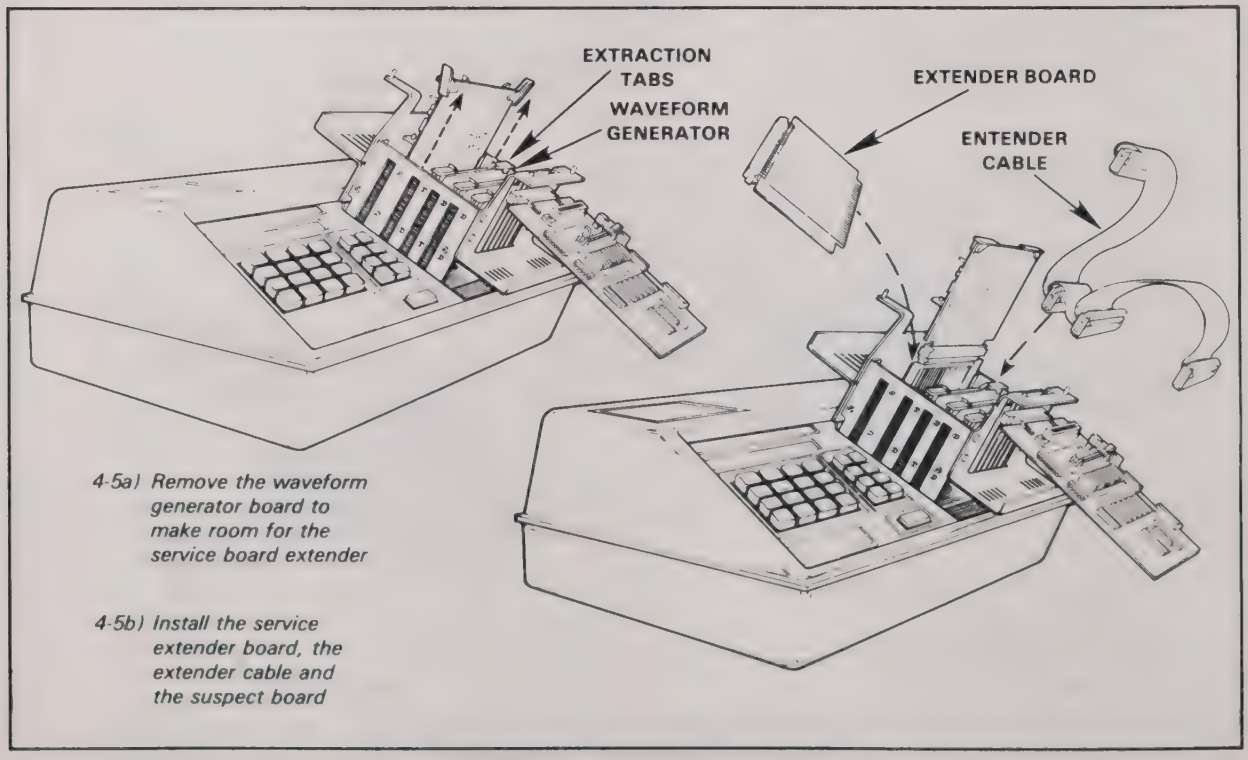

EXTRACTION TABS
WAVEFORM GENERATOR
EXTENDER BOARD
ENTENDER CABLE
4-5a) Remove the waveform generator board to make room for the service board extender
4-5b) Install the service extender board, the extender cable and the suspect board

Figure 4-5. Troubleshooting Equipment Setup

1. Check to be sure the LogicPak™ and adapters are properly installed in your programmer (see section 2.4).
2. Check to be sure the power cord is properly connected (see section 3.2).
3. Check the programmer mating connector (J1) for bent or broken pins. (Pin HH is intentionally shorter.)
4. Check the cards to be sure they are correctly installed in their connectors (section 4.4.1).
5. Check the ribbon cable to be sure it is properly inserted in the connectors (section 4.4.1).
6. Check the programmer power supplies for proper voltage output levels (see programmer manual).
7. If steps 1 through 5 fail to isolate the problem, contact your local Data I/O Service Center.

4.4.3 LogicPak™ Failure

The following steps should be performed if a device will not program at all or if error messages are displayed. After completing each step, determine whether the problem still exists.

1. Check that the family and pinout codes are correct for the device and that the device is being inserted in the correct socket (the one with the illuminated LED).
2. If possible, try to program a known good device to determine if it is a hardware or device problem.
3. Check to be sure the LogicPak™ and adapters are properly installed.
4. Check the programmer mating (J1) connector for bent or broken pins. (Pin HH is intentionally shorter.)
5. Check the LogicPak™ cards to be sure they are correctly installed in their connectors (section 4.4.1).
6. Check to be sure the ribbon cable is correctly and properly inserted in the connectors.
7. Perform a complete calibration noting any measurements falling outside of the indicated limits. Refer to the calibration section in each adapter manual, as well as the Circuit Description (section 5) and the schematics in this manual to attempt to isolate the problem.
8. Perform waveform observations and note any discrepancies. Referring to the Circuit Description and the schematics may be helpful in isolating the problem.
9. Refer to the section on limitations of the Logic Fingerprint™ test in the P/T adapter manual to see whether any of those conditions apply to the problem at hand. A Logic Fingerprint™ test that gives multiple test-sum for a single device is an indication of an oscillating condition.
10. If steps 1 through 9 fail to isolate the problem, contact your local Data I/O Service Center.

4.4.4 LogicPak™ Assembly

To remove the troubleshooting equipment and reassemble the LogicPak™, do the following steps:

1. Lift up the extraction tabs on the cable connector on the pin driver board and the three boards in the card cage.
2. Remove the service cable from the LogicPak™.
3. Unplug the board from the connector on the service extender board.
4. Unplug the service extender board from the motherboard.
5. Using the flat surfaces of the extraction tabs, gently but firmly push the waveform generator board along the guides until it locks into the J2 connector on the motherboard; the extraction tabs will lock into position when the board is properly inserted.

NOTE

Because of mechanical restrictions the waveform generator board can only be inserted into the J2 or J3 connector on the motherboard. The other three boards are interchangeable.

6. Gently, but firmly insert the normal interconnect cable into the connectors on the circuit boards.

NOTE

The extraction tabs will move back to their locked positions when the cable is locked into the connector.

7. Unplug the pin driver board from the bus extender board.
8. Rotate the pin driver board back 180° (the opposite of what is shown on figure 4-4) and insert the interconnect cable into the connector on the pin driver board.
9. Plug the pin driver board into the bus extender board as shown in figure 4-4.
10. Lower the LogicPak™ cover onto the assembly.
11. Replace the four allen screws and four phillips head screws to secure the cover in place.

SECTION 5

CIRCUIT DESCRIPTION

5.1 INTRODUCTION

This section defines the functions of the principal hardware components of LogicPak™. Each circuit-board assembly is depicted by a block diagram accompanied by a written description.

5.2 GENERAL ARCHITECTURE

The principal components, layout, and corresponding part numbers of the LogicPak™ are illustrated in the block diagram and are shown in figure 5-1. The LogicPak™ is controlled by the programmer's extended processor bus (J6 in figure 5-2) through the programmer's mating connector.

LogicPak™ adapters interface with the LogicPak™. When these adapters are installed, the LogicPak™ provides all the specific hardware and firmware required to support numerous families of programmable logic. Support provided includes methods of specifying the logic device design, fuse programming and verification, and functional testing.

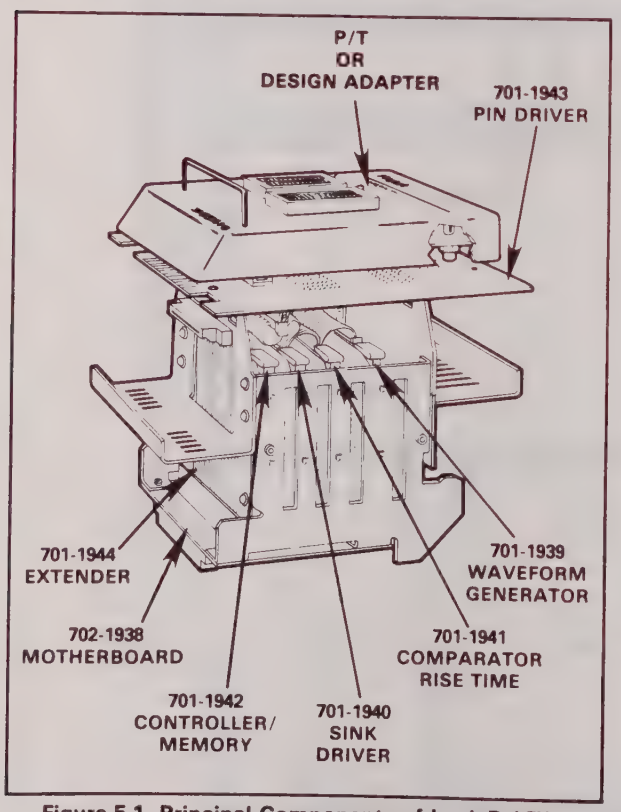


Figure 5-1. Principal Components of LogicPak™

5.3 COMPONENT LAYOUT

Each board is described, and block diagrams are presented in sections 5.3.1 through 5.3.7. Schematics are provided at the back of this manual.

5.3.1 MOTHERBOARD

The motherboard (702-1938) accepts the programmer processor bus and power supplies and routes them to the five 72-pin connectors. It also serves as the interconnect between the five connectors for signals generated on the plug-in cards.

5.3.2 EXTENDER

The extender board (701-1944) is installed in one of the five motherboard connectors and extends motherboard pins up to the top-mounted pin driver board (701-1943 in figure 5-2).

5.3.3 CONTROLLER/MEMORY

The controller/memory board (701-1942 in figure 5-2) has two functions: 1) it provides high-speed control operations that are not obtainable under processor operating speed, and 2) it contains the address decode circuitry that specifies the memory map for the base LogicPak™ and adapters. The block diagram for the controller/memory board is in figure 5-3.

Control function. A high-speed controller is necessary to supplement the programmer processor because of the large quantity of serial data that must be read during programming, verifying, and testing of logic devices; see the controller portion of the block diagram (figure 5-3).

A free-running crystal-controlled oscillator produces a 4-MHz clock pulse that synchronizes the on-card operations of the control counter, vector counter, synchronize register, and control register. Its output is routed to other boards via the motherboard.

The mode register stores instructions from the processor that are presented to the controller and test PROM; these instructions describe which high-speed cycle the controller should perform. During each cycle, the control counter addresses the controller and the test PROM. Their decoded output function determines what control states should exist for that cycle. These states, which are in the form of registered control lines, are routed to other cards to control the input and output register operations. Each cycle is executed by output from the synchronize register. This output synchronizes the processor operations to the controller cycles. This synchronization is a handshake, busy/ready type operation.

At the end of each cycle, data serially stored (under controller operation) in the read register are available for interrogation by the processor. These data represent: 1) logic device fuse states after fuse-related operations, 2) device output data during functional test operations, or 3) the Logic Fingerprint™ test.

During the Logic Fingerprint™ test, a vector counter tallies 128,000 cycles and then outputs a signal that tells the processor that one Logic Fingerprint™ test cycle has been completed. The test can then end or continue, based on user-selectable options. The test options range from Ø (no test performed) up to 99 cycles.

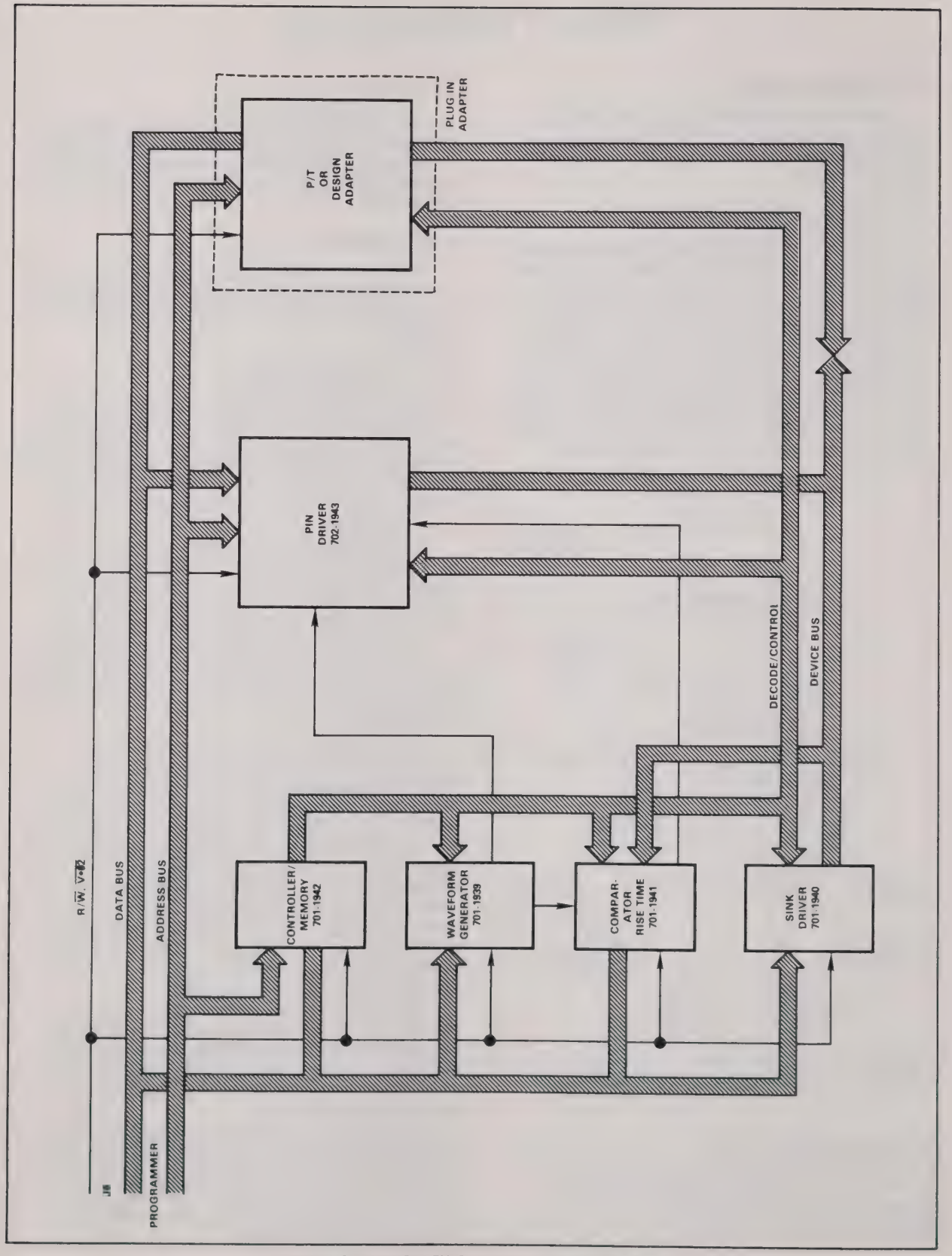


Figure 5-2. LogicPak™ Electronics Block Diagram

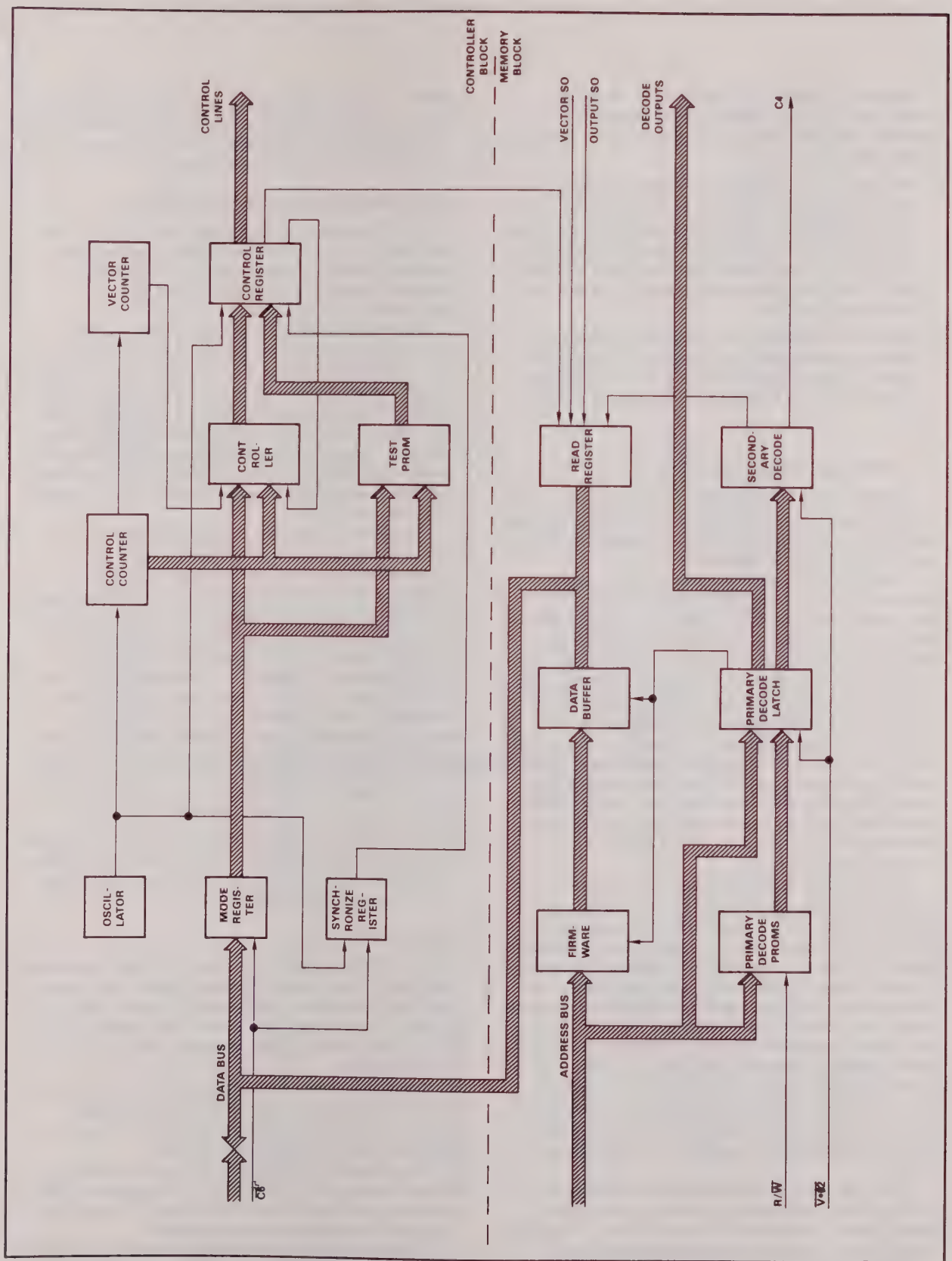


Figure 5-3. Controller/Memory (701-1942) Block Diagram

Memory function. The two parts of the memory function are: 1) primary decode, and 2) storage of LogicPak™ firmware; see the memory portion of the block diagram (figure 5-3).

1. **Primary decode.** The decode PROMs and decode latch allocate the available 16k bytes of processor memory to LogicPak™ functions. Where necessary, the output from the decode latch is routed to the secondary decode on other boards. Decode outputs are timed with V•Ø2 such that the selected locations are strobed to the data bus only during V•Ø2 time.
2. **LogicPak™ firmware.** The low-order 8k bytes of the 16k byte available space are allocated for LogicPak™ firmware. This firmware is stored in an 8k byte PROM whose outputs are buffered and presented to the data bus.

5.3.4 WAVEFORM GENERATOR

The waveform generator (701-1939 in figure 5-1) selects specific voltage levels required to perform logic device fuse read, fuse program, and functional test operations; see figure 5-4 for the block diagram.

The three major power supplies of the waveform generator are: V_{CC} supply, CE supply, and BIT supply. Each supply is under software control via a DAC (digital-to-analog converter). The DAC voltage output levels are obtained by presenting data via the data latch to the DAC inputs. This latch provides buffering and the timing required to store data. A common DAC reference voltage is generated from the DAC reference voltage circuit.

Each DAC output is routed to a power amp to provide for the high current requirements of the supplies. The power amps are op-amps with their positive voltage rails supplied through current sense detectors. The output rise and fall times are fixed by the slewing rate of the op-amp and are routed to the rise time section of the comparator/rise time board.

The current sense detectors have two selectable ranges. The ranges are controlled by the values stored in three bits of the control register. When activated, these detectors set the reset memory, which in turn kills all supply outputs by shutting down the DAC reference. The state of the reset memory is gated onto the data bus, and, if the memory state is set when inspected, an overcurrent error is flagged. The power-on reset circuit sets the reset memory when the programmer is powered up. When a device-related operation is selected, this memory is cleared by an initialize routine and remains cleared until an overcurrent condition sets the memory or the programmer is powered down. The current sense integrator smooths the transient overcurrent pulses, thereby avoiding false overcurrent errors.

Other functions of the control register are scope synchronization and provision of control lines for the adapter. These lines control the adapter LEDs, the backward device test circuitry, and the special clamping levels.

The comparator reference supply is identical in operation to the three other supplies except it does not require (or incorporate) a power amplifier and current sense stage. Its output is routed to the comparator section of the comparator/rise time board and is used as a programmable voltage reference level for the comparators.

5.3.5 COMPARATOR/RISE TIME BOARD

The comparator/rise time board (figure 5-5) has two functions: 1) it monitors the logic device pins and the waveform generator outputs, and 2) it converts the waveform outputs into switchable signals with programmable rise time control.

Rise time function. The switch control register determines which rise times are selected and which switches are to be activated. Each rise time control circuit consists of a current source (one common current value for all sources) driving the base of its switch transistor to charge the base capacitance. When the bases are released, a linear ramp is generated that is truncated at the switch supply level. The slew (dV/dt) rate is dependent on the value of capacitance switched to ground by the control register and the value of the current source.

The switch outputs and the nonswitched V_{CC} sense line are connected to the switch comparator. This comparator allows the switch and V_{CC} functions to be output as status to the status gate. Controller-ready status along with the switch/V_{CC} status are monitored on the data bus. Error codes are displayed if the switches or V_{CC} are not functional during the LogicPak™'s self-test.

Comparator function. The comparator monitors the pins of the logic device and compares them against the comparator voltage reference level. The comparators support the high-level program voltages and the TTL read levels of the device. Their outputs (device data) are stored in the output registers that, in turn, output data over the serial output (SO) line to the read register on the controller board. The register load and serial shift operations are synchronized to the controller operations by the control line (S/L) and oscillator (CP). As previously explained in section 5.3.3 (control function), this is how the logic device fuse and functional status are read.

5.3.6 PIN DRIVER

The pin driver (702-1943 in figure 5-2) board serves as a receptacle for the LogicPak™ socket adapters and contains the source driver circuitry that routes the switch outputs from the rise time board (BIT switch 1, BIT switch 2, CE switch) to the socket adapter receptacle. See figure 5-6 for the block diagram.

The source drivers are PNP transistor switches driven by current sources that are controlled by the source registers. The current sources provide a constant base drive to the PNP switches at all switch voltages.

The data latch provides the buffering and the timing required to store data in the CMOS source registers. This data storage will control the routing of the BIT switch and CE switch to the socket adapter receptacle.

The pin driver board also connects the sink driver board outputs and the extended processor bus extender board outputs to the socket adapter receptacle.

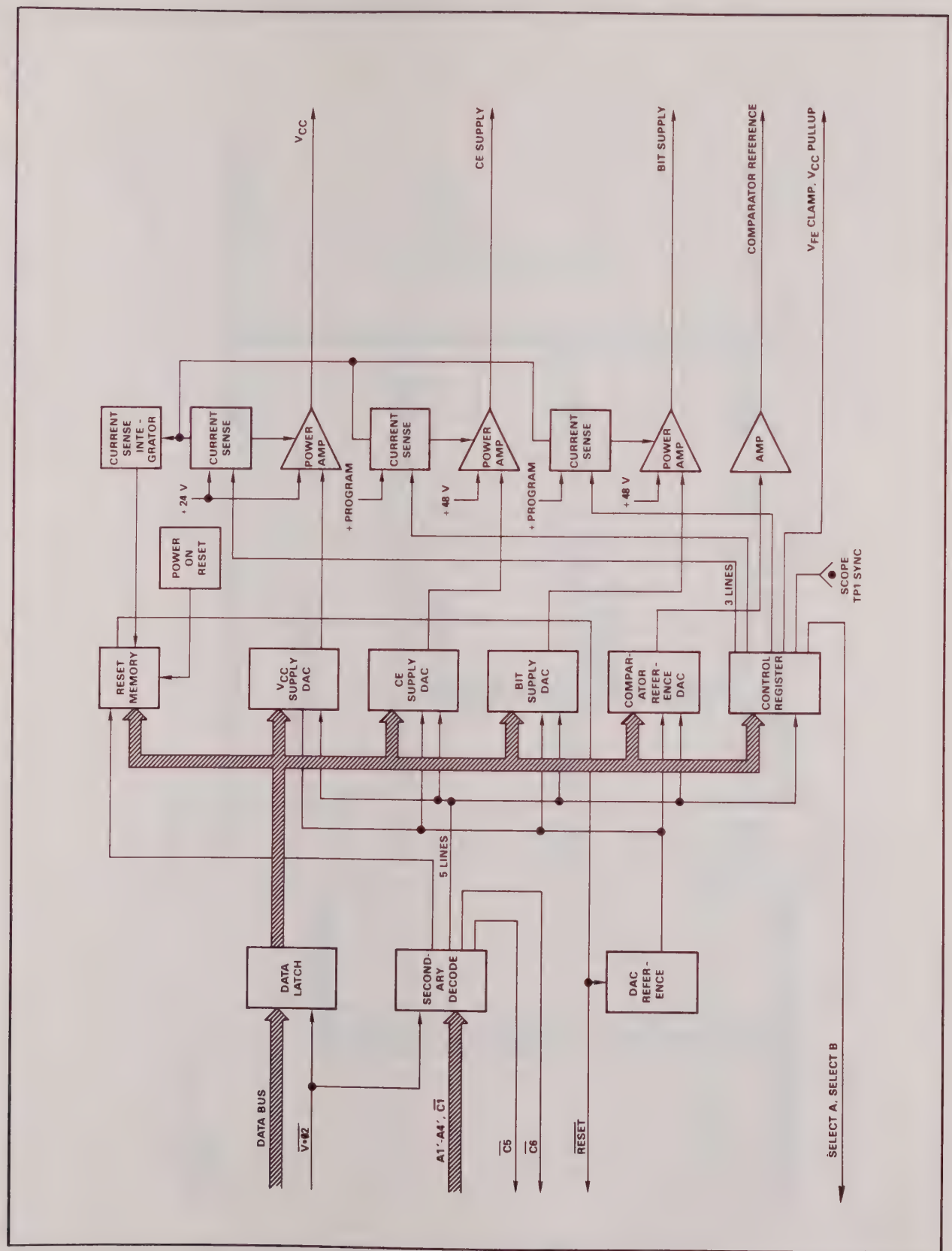


Figure 5-4. Waveform Generator (701-1939) Block Diagram

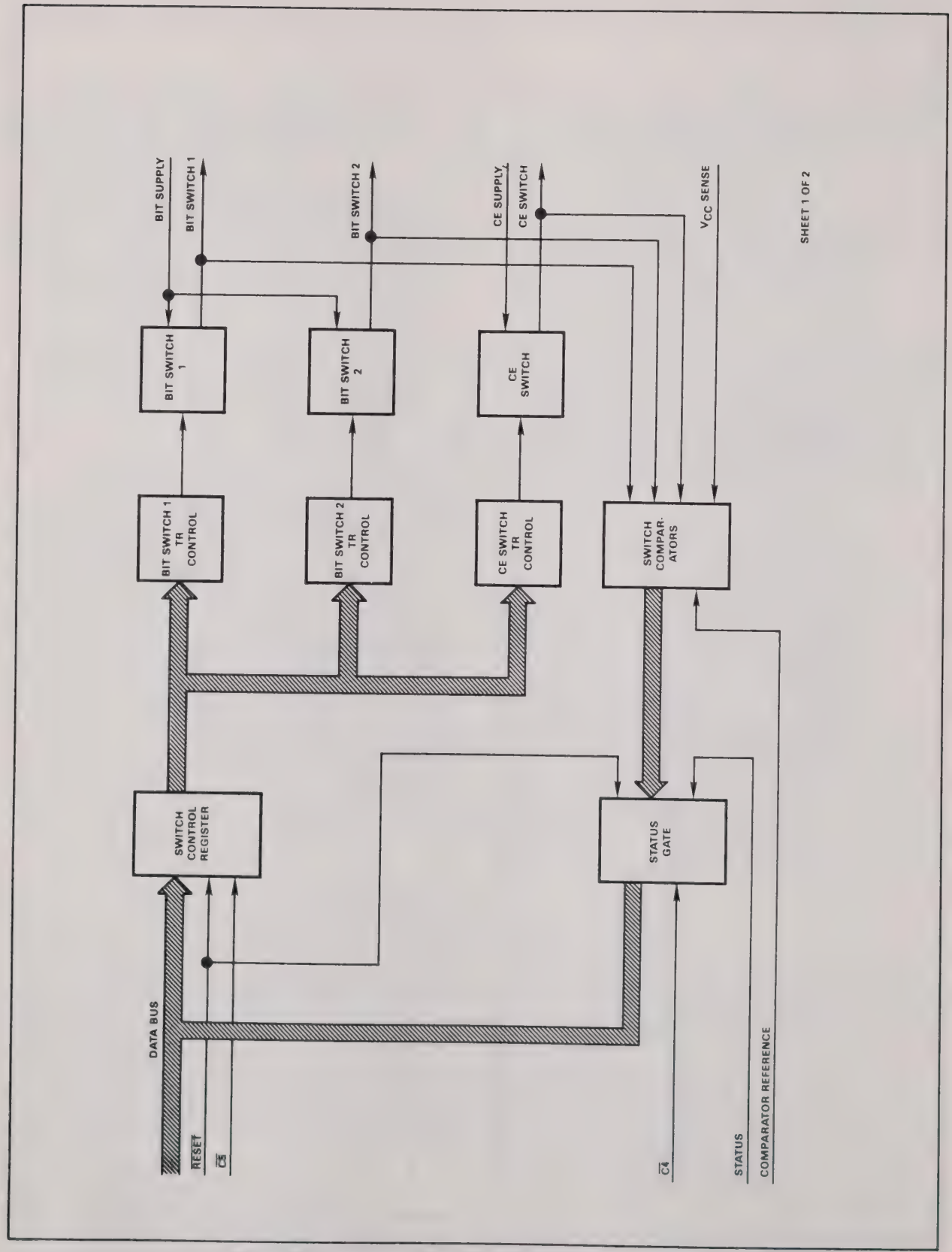


Figure 5-5. Comparator/Rise Time (701-1941) Block Diagram

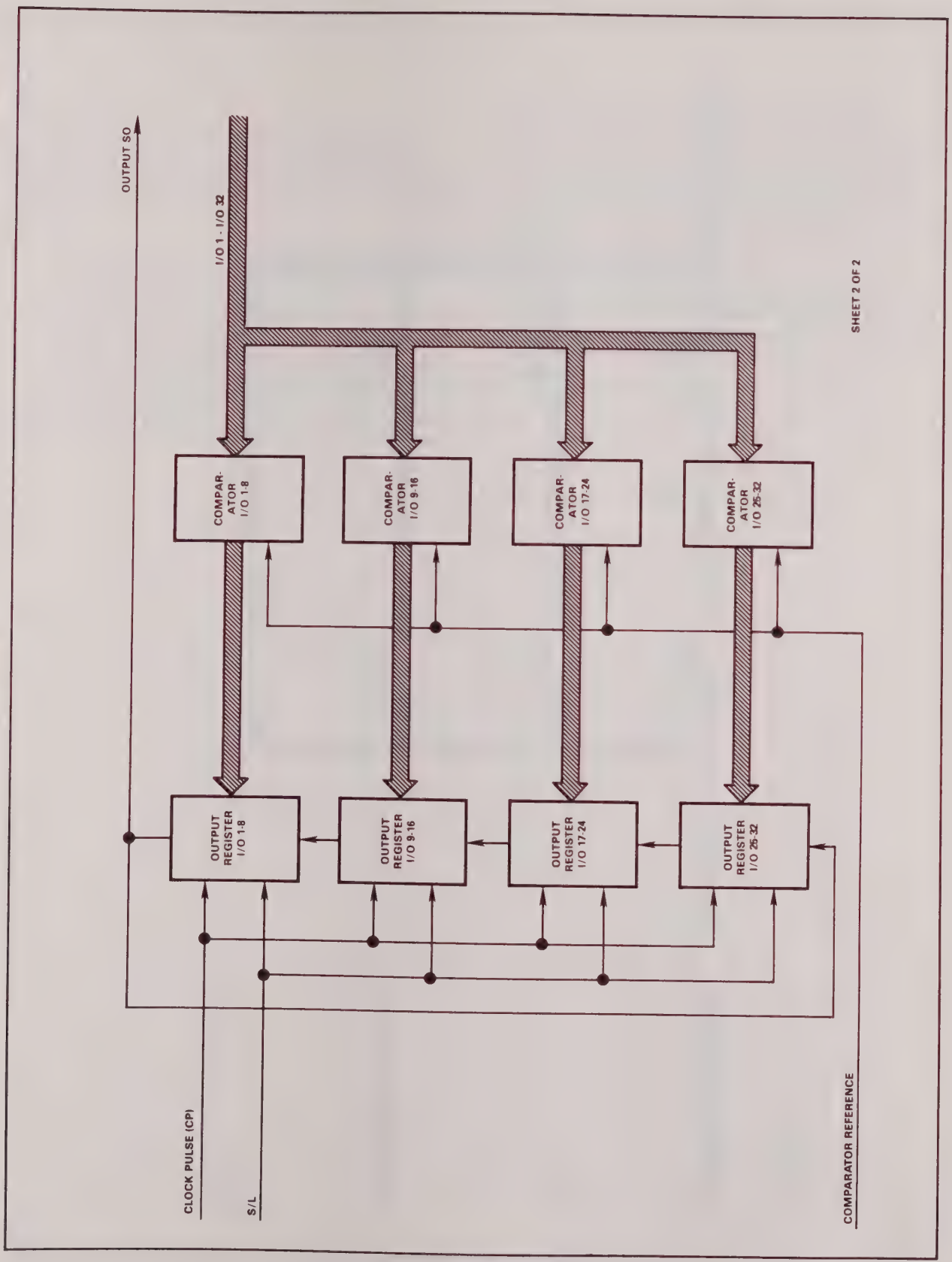


Figure 5-5. (Continued)

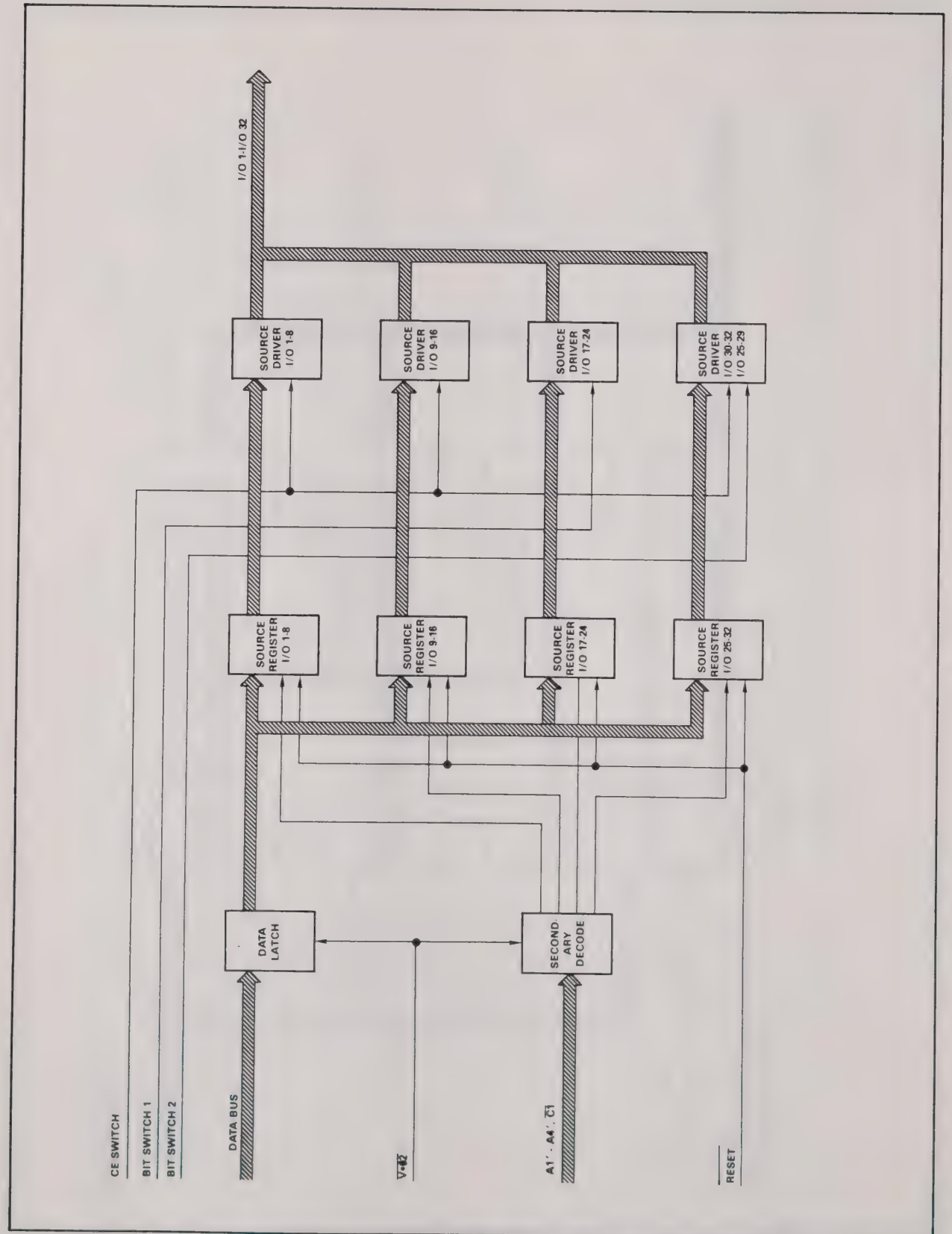


Figure 5-6. Pin Driver (702-1943) Block Diagram

5.3.7 SINK DRIVER

The sink driver board (figure 5-7) provides the TTL levels that are necessary to read and program device fuses and perform function testing. The sink drivers are high-voltage noncommitted collector drivers. Their output drive is limited so that during a functional test, the logic device can override.

The input registers control the level of the sink drivers. These registers are under software control for fuse and structured test operations and are under controller board control for Logic Fingerprint™ test operations.

In Logic Fingerprint™ test operation, the registers are preloaded by using the starting vector specified. The Logic Fingerprint™ test cycle then begins, starting a serial shift operation of the registers with serial data input from the parity generator. Following 32 clock cycles (one controller cycle), the next Logic Fingerprint™ test vector is presented, and the register will halt until the logic device pins are tested. This procedure will continue until the controller board signals that the Logic Fingerprint™ test is then read by additional shift cycles with the serial data input being that of the register serial output (recirculate). The shift modes are controlled by control lines IRC and CP (see figure 5-7) and the data recirculation path is enabled by control line RICR.

The CP and CE multiplexer changes control of up to four sink drivers from input register control line to controller board control. This alternate control is necessary during the Logic Fingerprint™ test to provide special consideration to logic devices that have dedicated clock, reset, preset, and enable pins.

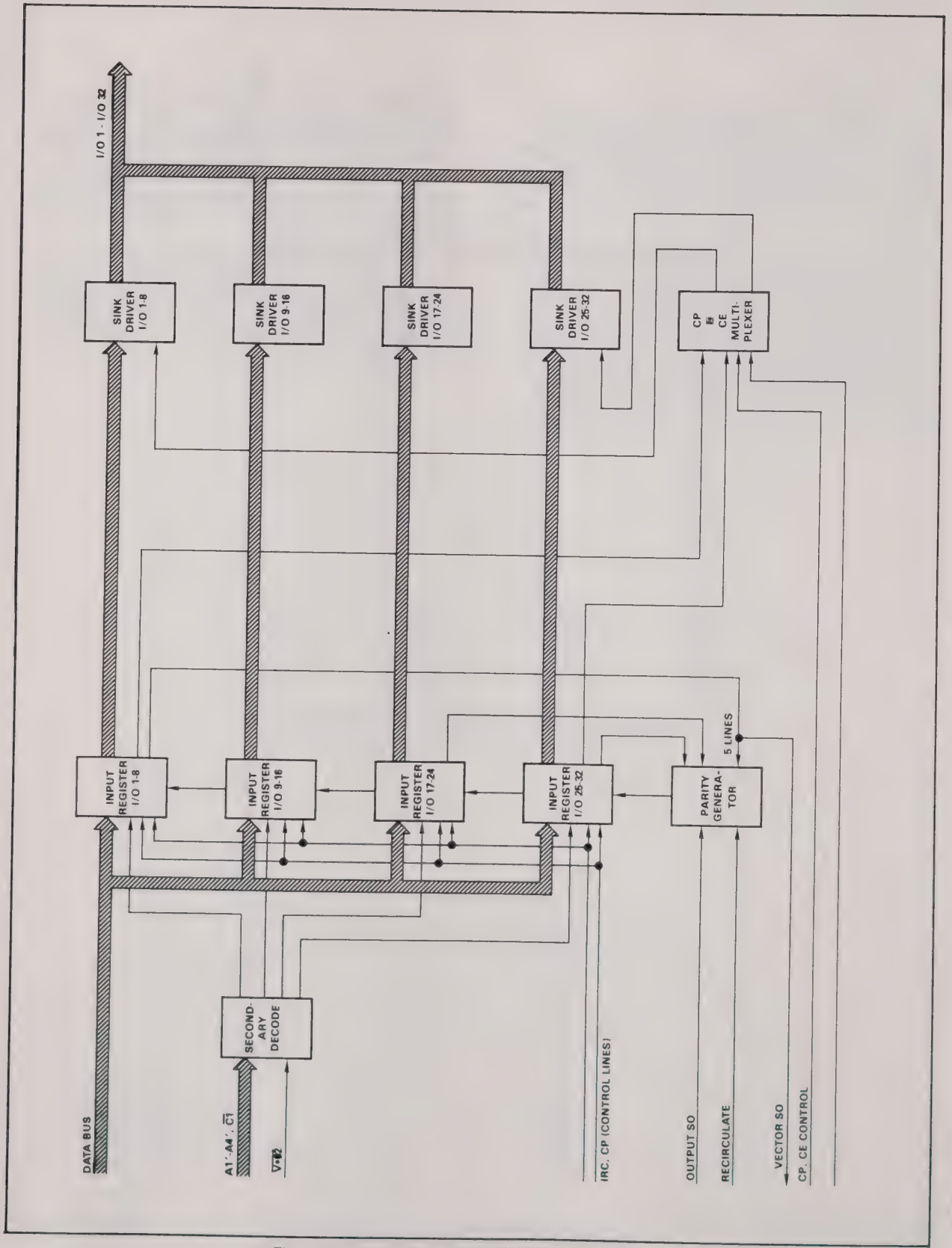


Figure 5-7. Sink Driver (701-1940) Block Diagram

APPENDIX A

STANDARD DATA TRANSFER FORMAT BETWEEN DATA PREPARATION SYSTEM AND PROGRAMMABLE LOGIC DEVICE PROGRAMMER

This document defines a format for the transfer of information between a data preparation or storage system and a device programmer. This format provides for, but is not limited to, the transfer of fuse, test, identification, and comment information in an ASCII representation. This format defines the "intermediate code" between device programmers and data preparation storage systems.

NOTE

This is Data I/O's implementation of the JEDEC (Joint Electron Device Engineering Council) standard (JC-42.1-81-62).

Copyright 1983

Data I/O Corporation
10525 Willows Road N.E./C-46
Redmond, WA 98052
(206) 881-6444

Version 1.0 Feb. 22, 1983

Michael Holley

TABLE OF CONTENTS

SECTION 1. INTRODUCTION

1.1 BNF RULES A-1

1.2 BNF DEFINITIONS A-1

SECTION 2. TRANSMISSION PROTOCOL

2.1 DESIGN SPECIFICATION A-2

2.2 TRANSMISSION CHECK-SUM A-2

2.3 FIELDS A-2

SECTION 3. FORMAT FIELD DEFINITIONS

3.1 DEVICE TO BE PROGRAMMED A-3

3.2 FUSE INFORMATION A-3

3.3 STRUCTURED FUNCTIONAL TEST INFORMATION A-3

3.4 OPTIONAL INFORMATION A-4
3.4.1 Logic Fingerprint™ A-4
3.4.2 Security Fuse A-4

SECTION 4. OTHER RULES

4.1 TRANSPORTABILITY A-5

4.2 RESTRICTIONS AND VARIATIONS A-5
4.2.1 Variations From Present JEDEC Standard A-5
4.2.2 Variations For 303A LogicPak™ A-5

SECTION 1

INTRODUCTION

The Backus-Naur Form (BNF) is used in this document to define the format syntax.

1.1 BNF RULES

"::=" denotes "is defined as".

Characters enclosed by single quotes are literals.

Parentheses enclose indentifiers.

Square brackets enclose optional items.

Braces (curly brackets) enclose a repeated item. The item may appear zero or more times.

A vertical bar separates alternative items.

A repeat count is given by a ":n" suffix. For example, a six digit number would be defined as (number) ::= (digit) :6.

Example

The English description of a person's name may be as follows:

> The full name consists of an optional title followed by a first name, a middle name, and a last name. The person may not have a middle name or may have several middle names. The titles consist of : Mr., Mrs., Ms., Miss, and Dr.

The BNF definition would be:

(full name) ::= [(title)] (f. name) { (m. name)}
(l. name)

(title) ::= 'Mr.' | 'Mrs.' | 'Ms.' | 'Miss' | 'Dr.'

1.2 BNF DEFINITIONS

(digit) ::= '0' | '1' | '2' | '3' | '4' | '5' | '6' | '7' | '8' | '9'

(hex-digit) ::= (digit) | 'A' | 'B' | 'C' | 'D' | 'E' | 'F'

(binary-digit) ::= '0' | '1'

(number) ::= (digit) {(digit)}

(del) ::= (space) | (carriage return) | (line feed)

(delimiter) ::= (del) {(del)}

(printable character) ::= (ASCII 20 hex ... 7E hex)

(control character)::= (ASCII 00 hex ... 1F hex)
| (ASCII 7F hex)

(STX) ::= (ASCII 02 hex)
(ETX) ::= (ASCII 03 hex)
(carriage return) ::= (ASCII 0D hex)
(line feed) ::= (ASCII 0A hex)
(space) ::= (ASCII 20 hex) | ' '

(valid character) ::= (printable character)
| (carriage return) | (line feed)

(field character) ::= (ASCII 20 hex ... 2A hex)
| (ASCII 2C hex ... 7E hex) | (carriage return) | (line feed)

FORMAT FIELD DEFINITIONS

3.1 DEVICE TO BE PROGRAMMED

The device field defines the specific device being programmed. Data I/O uses this field to transmit the family and pinout codes.

BNF Syntax

(device) ::= 'D' (family-pinout code) '*'

(family-pinout code) ::= (hex-digit) :4

Example

D9501*

3.2 FUSE INFORMATION

Each device fuse link is assigned a decimal number. The fuse may have two possible states: a zero to specify a low resistance link and a one to specify a high resistance link.

The fuse information is presented in three fields: the default state (F), the fuse list (L), and the fuse check-sum (C). The default state and fuse check-sum fields are optional.

The "F" field is used to define the states of links not explicitly defined in the "L" fields. If the default state is not used, all device links must be defined in the "L" fields.

The "L" field can be any convenient length, and any number of "L" fields can be used. If the state of a link is specified 2 or more times, the last state replaces the preceding entries. The number following the "L" indicates the starting fuse number for the data string.

The link information check-sum is computed by performing a 16 bit addition of 8 bit words constructed from the specified state of each link in the device. The unused bits in the last word are set to zero. The 8 bit words are defined as shown in the following diagram.

word 00	msb							lsb
Link No.	7	6	5	4	3	2	1	0
word 01	msb							lsb
Link No.	15	14	13	12	11	10	9	8

---

word 62	msb							lsb
Link No.	-	-	-	-	499	498	497	496

BNF Syntax

(fuse information) ::= [(default state)] (fuse list) {(fuse list)} [(fuse check-sum)]

(default state) ::= 'F' (binary-digit) '*'

(fuse list) ::= 'L' (number) (delimiter) {(binary-digit) [(delimiter)]} '*'

(fuse check-sum) ::= 'C' (hex-digit):4 '*'

Example

F0*L0 01010101* L0008 01010111*L1000 0101*CF3BA*

Example

L0200 0110101010101010101101011101011010001001 0010010*

3.3 STRUCTURED FUNCTIONAL TEST INFORMATION

This field specifies one of several test conditions for each pin of a device. A test vector is defined as N test conditions where N is the number of pins on the device.

Test Conditions

- 0 - Drive input low
- 1 - Drive input high
- 2-9 - Drive input to supervoltage #2-9
- C - Drive input low, high, low
- K - Drive input high, low, high
- N - Power pins and outputs not tested
- L - Test output low
- H - Test output high
- Z - Test input or output for high impedance
- F - Float input or output

The C and K driving signals are presented after the other inputs are stable. The L, H, and Z tests are performed after all inputs have stabilized including C and K.

The test vectors will be applied to the device under test in numerical order. If any vectors are specified 2 or more times the data in the last vector replaces the previous data. There is no default test vector.

The optional "P" field is used to specify the correspondence of the test conditions to the device pin numbers. If the "P" field is not used, the first condition will be applied to pin 1, the second to pin 2, and so on through pin N.

BNF Syntax

(function test) ::= [(pin list)] (test vector) {(test vector)}

(pin list) ::= 'P' (pin number):N '*'

(pin number) ::= (delimiter) (number)

N ::= number of pins on device

(test vector) ::= 'V' (number) (delimiter) (test condition):N '*' (test condition) ::= (digit) | 'C' | 'F' | 'H' | 'K' | 'L' | 'N' | 'Z'

(reserved condition) ::= 'A' | 'B' | 'D' | 'E' | 'G' | 'I' | 'J' | 'M' | 'O' | 'P' | 'Q' | 'R' | 'Y' | 'Z'

Example

```
P 11  12  13  14  15  16  17  18  19  20
10   9   8   7   6   5   4   3   2   1*

V0001 C01010101NHLLLHHLHLN*
V0002 C01011111NHLLHLLLHLN*
V0003 C10010111NZZZZZZZZZZN*

V4 C01010100NFLHHLFFLLN*
```

3.4 OPTIONAL INFORMATION

Additional option fields may be defined using the letters G, S, R, M, Q, and T. Each field must begin with one of the above letters and be terminated with an asterisk (*). No other restrictions are applied. Therefore, multiple letters could be used to create any number of option fields.

BNF Syntax

(option field) ::= (option ident.) {(field character)} '*'

(option ident.) ::= 'G' | 'S' | 'R' | 'M' | 'Q' | 'T'

Example

```
Q*
MFG Acme Semiconductor*
M:1234*
```

Data I/O uses four optional fields; three for the Logic Fingerprint™ test (subsection 3.4.1), and a security fuse field (subsection 3.4.2).

3.4.1 LOGIC FINGERPRINT™

The 'S' field defines the starting vector for the Logic Fingerprint™ test. The possible states are 0 (TTL low) and 1 (TTL high). The 'R' field contains the resulting vector or test-sum. The 'T' field denotes the number of test cycles to be run.

BNF Syntax

(starting vector) ::= 'S' (test condition):N '*'

(resulting vector) ::= 'R' (hex-digit):8 '*'

(test cycles) ::= 'T' (number) '*'

N ::= number of pins on device

Example

```
S01000100001110001111011O*
R5BCD34A7*
T01*
```

3.4.2 SECURITY FUSE

The security fuses of certain logic devices may be enabled for programming by sending a "1" in the "G" field.

BNF Syntax

(security fuse) ::= 'G' (binary-digit) '*'

Example

```
G1*
```

SECTION 4

OTHER RULES

4.1 TRANSPORTABILITY

All receiving machines should have a "kernel" mode to ignore all optional fields so that the actual programming data will be transportable. For example, as allowed in the format, optional fields can be sent to specify additional checksums. A receiving machine in the "kernel" mode could ignore this information yet receive the link information required to program the device.

If the optional "F" field is used to avoid transmitting link data, transportability could be lost. Therefore, whenever practical, data should be transmitted for all links of the device.

BNF Syntax

(kernel) ::= (STX) (design spec) (min. fuse information) (ETX) (xmit check-sum)

(design spec) ::= {(field character)} '*'

(min. fuse information) ::= (fuse list) {(fuse list)}

4.2 RESTRICTIONS AND VARIATIONS

4.2.1 VARIATIONS FROM PRESENT JEDEC STANDARD

a) The delimiter after the link number or the vector number may be any combination of carriage returns, line feeds, and/or spaces.

b) The check-sum and sum-check are renamed fuse check-sum and xmit check-sum respectively.

4.2.2 VARIATIONS FOR 303A LOGICPAK™

a) The link number ("L" field) and the vector number ("V" field) must be 4-digit numbers.

b) The delimiter after the link number and the vector number must include a space.

c) Spaces are not allowed between the terminating asterisk of one field and the identifier of the next field.

d) The security fuse ("G" field) requires a space after the binary digit.

e) The kernel mode is not implemented.

APPENDIX B

REFERENCE MATERIAL

Glossary

address field. Optional set of control characters in a data translation format. It defines the address of the next data byte.

address offset. A four-digit hexadecimal value subtracted from addresses on input and added to addresses output from the programmer. The result is added to the begin device address or begin RAM address as applicable.

begin device address. The first device address from which or to which data are being transferred.

begin RAM address. The first address of the programmer data RAM from which or to which data are to be transferred.

blank check. A test performed by a programmer to detect the presence of any programmed bits. A device with no programmed bits is "blank."

block size. The hexadecimal number of bytes to be transferred in a data transfer.

configuration number. A four-digit hexadecimal number that identifies the software revision level of the programmer.

data translation format. Form in which the translator software accepts input data. Also the form for data output by the unit.

default value. The value the unit uses for a parameter unless the operator specifies another value.

device. Any PROM, EPROM, MOS PROM, or programmable logic array.

end code. Character in a data translation format that signals the completion of a data transfer.

error code. A code that signals specific errors to the operator.

family and pinout codes. Two-digit codes used by some Data I/O programming modules to identify programming variables including pinout, address limit and programming algorithms.

full duplex. Simultaneous, two-way independent transmission in both directions.

handshaking. The required sequence of signals for communication between two units. The I/O bus protocol for a unit defines its handshaking requirements. This is especially true for asynchronous I/O systems in which each signal requires a response to complete an I/O operation.

half duplex. Transmission alternately in either direction but not both directions simultaneously.

illegal-bit test. A test performed by a programmer to detect the presence of any programmed bits of incorrect polarity (illegal bits).

logic gate. Digital circuits with two or more logic inputs and one output, with a logic level determined by the logic levels present at the inputs. The basic logic gates are:

OR produces a 1 (hi) when one or more inputs are 1.
AND produces a 1 only when all inputs are 1.
NOR produces Ø (low) when any input is 1.
NAND produces a Ø only when all inputs are at different levels
EX-NOR produces a 1 only when the inputs are at the same logic level.

mode. A software routine in a machine, characterized by a specific automatic sequence of steps.

select function (also select code). A two-digit hexadecimal number used to specify data translation formats, serial interface operations, or certain RAM data manipulations.

start code. Character in a data translation format that signals the beginning of a data transfer.

sum-check. A summation of bits calculated according to the rules of simple addition and usually expressed as a four-digit hexadecimal number; any carry from the most significant bit or digit is discarded. A sum-check is used to verify the integrity of data transfers.

waveforms (programmable). The graphical representation of the timing and magnitude of programming pulses. If the programming waveforms are not kept within tolerance, programming yield is jeopardized.

Abbreviations

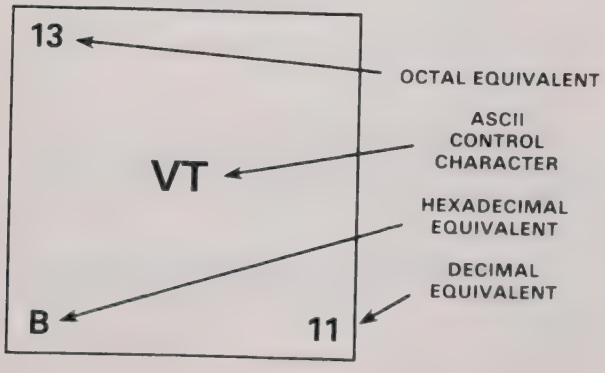
The following is a list of abbreviations commonly used in Data I/O Instruction manuals.

amp - amplifier or ampere
ASCII - American Standard Code for Information Interchange
CE - chip enable
CMOS - complementary metal oxide semiconductor
CP - clock pulse
CR - carriage return
CTRL - control
CTS - clear to send
DAC - digital-to-analog converter
dV/dt - delta voltage versus delta time
E - enable pulse derived from the 6802 microprocessor
EPROM - erasable, programmable read-only memory
ESC - escape
ETX - end of text
FPGA - field programmable gate array
FPLA - field programmable logic array
FPLS - field programmable logic sequencer
H&L - high and low
hex - hexadecimal
IFL - integrated fuse logic
I/O - input/output
JEDEC - Joint Electron Device Engineering Council
K - thousand
LED - light-emitting diode
LF - line feed
MHz - megahertz
No. - number
op-amp - operational amplifier
PAL - programmable array logic
PALASM - PAL assembler
PCS - program card set
PLDS - programmable logic development system
PROM - programmable read-only memory
P/T - programming/testing
RAM - random-access memory
Rev - revision
R/$\overline{W}$ - read/$\overline{write}$
S/L - control line
SO - serial output
STX - start of text
SYNC - synchronous
TP - test point
TTL - transistor-transistor logic
V•Ø2 - timing pulse, sum of E and VMA
V - volt
V_{CC} - power supply voltage applied to integrated circuits
VMA - valid memory address

ASCII CONTROL CHARACTERS

ACK	acknowledge
BEL	bell
BS	backspace
CAN	cancel
CR	carriage return
DC1	playback on, CNTL Q, X-ON
DC2	record on, CNTL R, PUNCH-ON, SOM
DC3	playback off, CNTL S, X-OFF
DC4	record off, CNTL T, PUNCH-OFF, EOM
DEL	delete, rubout
DLE	data link escape
EM	end of medium
ENQ	enquiry
EOT	end of transmission
ESC	escape
ETB	end of transmission block
ETX	end of text
FF	form feed
FS	file separater
GS	group separater
HT	horizontal tabulation
LF	line feed
NAK	negative acknowledge
NUL	null
RS	record separater
SI	shift in
SO	shift out
SOH	start of heading
STX	start of text
SUB	substitute
SYN	synchronous idle
US	unit separater
VT	vertical tab

Code Chart Key



Cross-Reference Chart of Number Bases

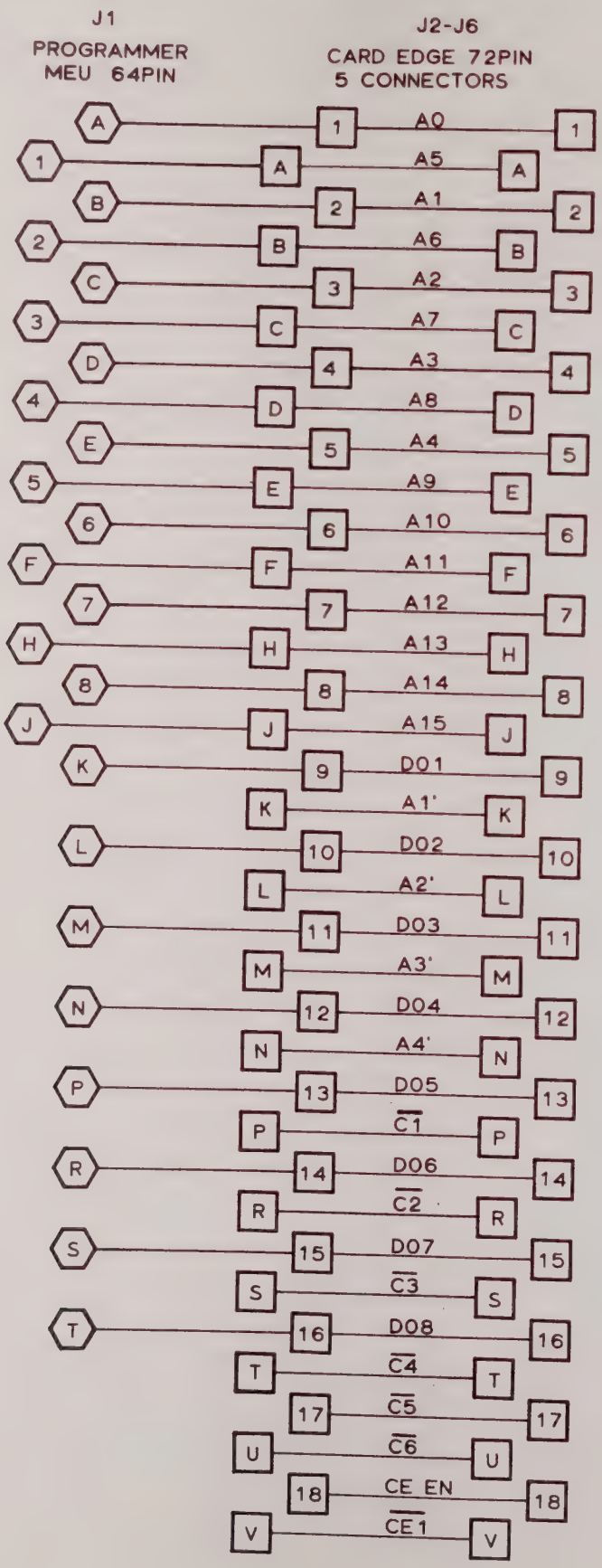
Binary	Octal	Hexadecimal	Decimal	Standard Abbreviation
0000	0	0	0	
0001	1	1	1	
0010	2	2	2	
0011	3	3	3	
0100	4	4	4	
0101	5	5	5	
0110	6	6	6	
0111	7	7	7	
1000	10	8	8	
1001	11	9	9	
1010	12	A	10	
1011	13	B	11	
1100	14	C	12	
1101	15	D	13	
1110	16	E	14	
1111	17	F	15	
0001 0000	20	10	16	
0010 0000	40	20	32	
0100 0000	100	40	64	
1000 0000	200	80	128	
0001 0000 0000	400	100	256	
0010 0000 0000	1000	200	512	
0100 0000 0000	2000	400	1,024	1K
1000 0000 0000	4000	800	2,048	2K
1100 0000 0000	6000	C00	3,072	3K
0001 0000 0000 0000	10000	1000	4,096	4K
0001 0100 0000 0000	12000	1400	5,120	5K
0001 1000 0000 0000	14000	1800	6,144	6K
0001 1100 0000 0000	16000	1C00	7,168	7K
0010 0000 0000 0000	20000	2000	8,192	8K
0010 0100 0000 0000	22000	2400	9,216	9K
0010 1000 0000 0000	24000	2800	10,240	10K
0100 0000 0000 0000	40000	4000	16,384	16K
1000 0000 0000 0000	100000	8000	32,768	32K
0001 0000 0000 0000 0000	200000	10000	65,536	64K

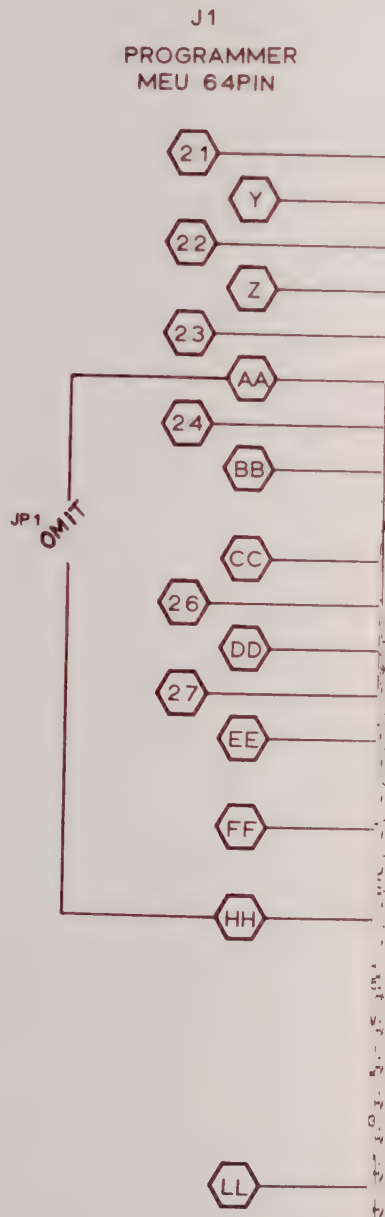
ASCII and IEEE Code Chart

BITS 4 3 2 1	7 6 5 = 0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
	CONTROL		NUMBERS AND SYMBOLS		UPPERCASE		LOWERCASE	
0 0 0 0	0 NUL 0 0	20 DLE 10 16	40 SP 20 32	60 0 30 48	100 @ 40 64	120 P 50 80	140 ` 60 96	160 p 70 112
0 0 0 1	1 SOH 1 1	21 DC1 11 17	41 ! 21 33	61 1 31 49	101 A 41 65	121 Q 51 81	141 a 61 97	161 q 71 113
0 0 1 0	2 STX 2 2	22 DC2 12 18	42 " 22 34	62 2 32 50	102 B 42 66	122 R 52 82	142 b 62 98	162 r 72 114
0 0 1 1	3 ETX 3 3	23 DC3 13 19	43 # 23 35	63 3 33 51	103 C 43 67	123 S 53 83	143 c 63 99	163 s 73 115
0 1 0 0	4 EOT 4 4	24 DC4 14 20	44 $ 24 36	64 4 34 52	104 D 44 68	124 T 54 84	144 d 64 100	164 t 74 116
0 1 0 1	5 ENQ 5 5	25 NAK 15 21	45 % 25 37	65 5 35 53	105 E 45 69	125 U 55 85	145 e 65 101	165 u 75 117
0 1 1 0	6 ACK 6 6	26 SYN 16 22	46 & 26 38	66 6 36 54	106 F 46 70	126 V 56 86	146 f 66 102	166 v 76 118
0 1 1 1	7 BEL 7 7	27 ETB 17 23	47 ' 27 39	67 7 37 55	107 G 47 71	127 W 57 87	147 g 67 103	167 w 77 119
1 0 0 0	10 BS 8 8	30 CAN 18 24	50 (28 40	70 8 38 56	110 H 48 72	130 X 58 88	150 h 68 104	170 x 78 120
1 0 0 1	11 HT 9 9	31 EM 19 25	51) 29 41	71 9 39 57	111 I 49 73	131 Y 59 89	151 i 69 105	171 y 79 121
1 0 1 0	12 LF A 10	32 SUB 1A 26	52 * 2A 42	72 : 3A 58	112 J 4A 74	132 Z 5A 90	152 j 6A 106	172 z 7A 122
1 0 1 1	13 VT B 11	33 ESC 1B 27	53 + 2B 43	73 ; 3B 59	113 K 4B 75	133 [5B 91	153 k 6B 107	173 { 7B 123
1 1 0 0	14 FF C 12	34 FS 1C 28	54 , 2C 44	74 V 3C 60	114 L 4C 76	134 \ 5C 92	154 l 6C 108	174 | 7C 124
1 1 0 1	15 CR D 13	35 GS 1D 29	55 - 2D 45	75 = 3D 61	115 M 4D 77	135] 5D 93	155 m 6D 109	175 } 7D 125
1 1 1 0	16 SO E 14	36 RS 1E 30	56 . 2E 46	76 > 3E 62	116 N 4E 78	136 ^ 5E 94	156 n 6E 110	176 ~ 7E 126
1 1 1 1	17 SI F 15	37 US 1F 31	57 / 2F 47	77 ? 3F 63	117 O 4F 79	137 ← 5F 95	157 o 6F 111	177 Rubout 7F 127
	Addressed Commands	Universal Commands	Listen Addresses		Talk Addresses		Secondary Addresses or Commands	

SCHEMATICS

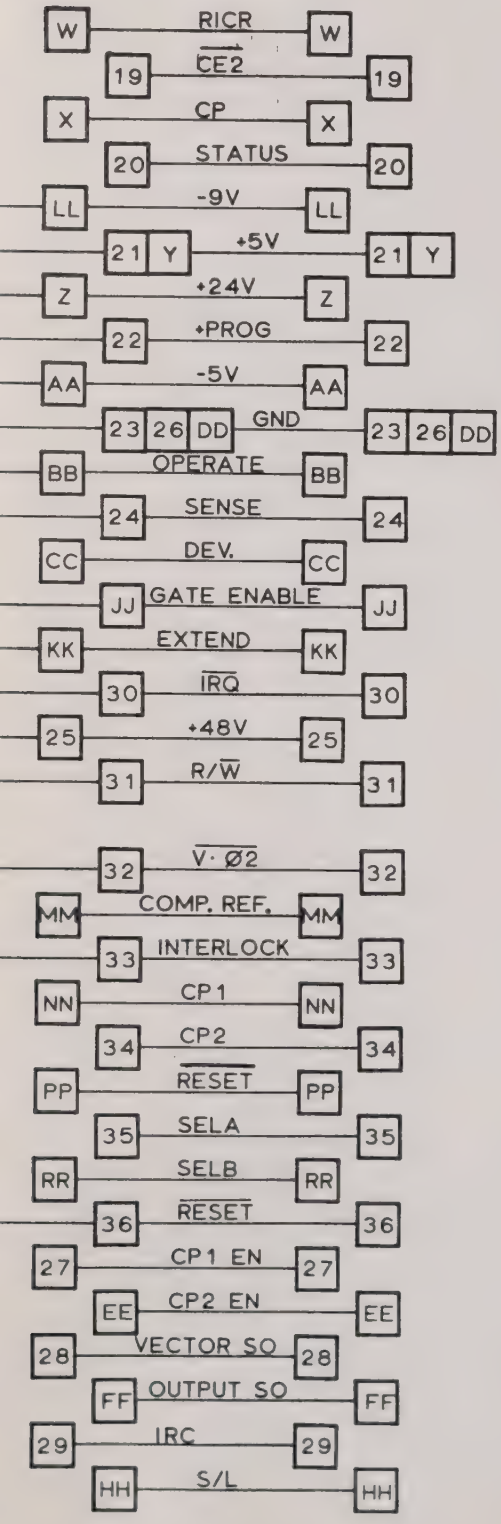
30-702-1938	Motherboard
30-701-1939	Waveform Generator
30-701-1940	Sink Driver
30-701-1941	T/Rise Comparator
30-701-1942	Controller/Memory
30-702-1943	Pin Driver
30-701-1944	Extender Board


J1
PROGRAMMER
MEU 64PIN
J2-J6
CARD EDGE 72PIN
5 CONNECTORS
A 1 A0 1
1 A A5 A
B 2 A1 2
2 B A6 B
C 3 A2 3
3 C A7 C
D 4 A3 4
4 D A8 D
E 5 A4 5
5 E A9 E
6 6 A10 6
F F A11 F
7 7 A12 7
H H A13 H
8 8 A14 8
J J A15 J
K 9 D01 9
K A1' K
L 10 D02 10
L A2' L
M 11 D03 11
M A3' M
N 12 D04 12
N A4' N
P 13 D05 13
P C1 P
R 14 D06 14
R C2 R
S 15 D07 15
S C3 S
T 16 D08 16
T C4 T
17 C5 17
U C6 U
18 CE EN 18
V CE1 V


J1
PROGRAMMER
MEU 64PIN
21
Y
22
Z
23
AA
24
BB
JP1
OMIT
CC
26
DD
27
EE
FF
HH
LL

REVISIONS					
LTR	DESCRIPTION	DR	CHK	APPR'D	DATE
A	RELEASE	JK			11/21/80
B	ECN 4691	CL			9/17/81
C	ECN #4785 3-83	SH		69	3/83

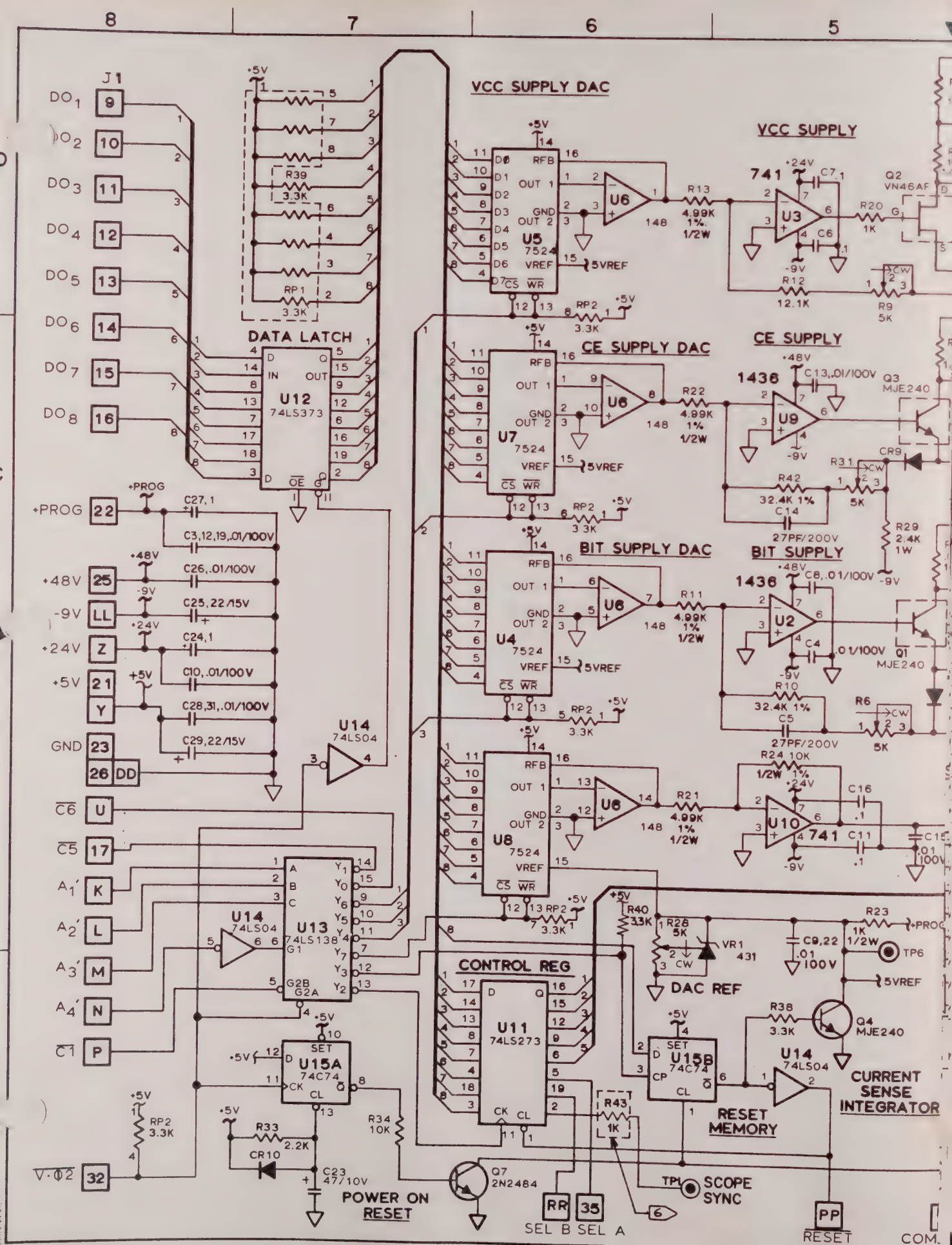
J2-J6
CARD EDGE 72PIN
5 CONNECTORS

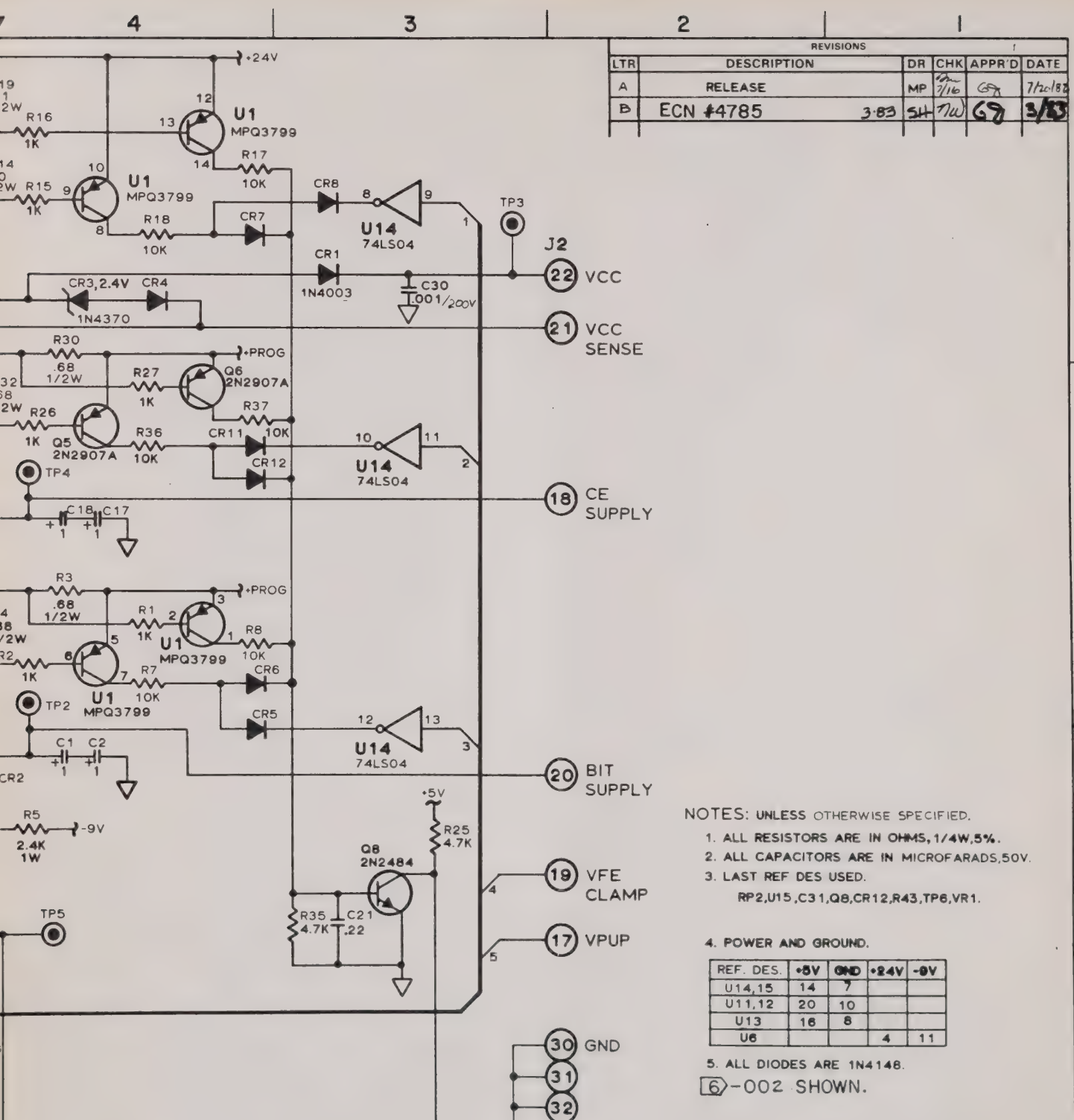


NOTES: UNLESS OTHERWISE SPECIFIED

1. LAST REFERENCE DESIGNATOR USED: JP1, J6

APPROVALS:		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES.		DATA I/O	
DSN. ENGR	DATE	TOLERANCES: UNLESS OTHERWISE SPECIFIED: XX ± XXX ± ANGULAR		TITLE SCHEMATIC DIAGRAM, MOTHER BOARD	
MFG. ENGR	DATE				
QUAL ASSUR	DATE	DRAWN BY: JAEHEE KIM	DATE 12/81	SIZE D	CODE INDENT NO 54193
ENGR MGR	DATE	CHECKED BY:	DATE	DRAWING NO 30-702-1938	
				SCALE	12-17-81 SHEET 1OF1


8
7
6
5
J1
DO1
DO2
DO3
DO4
DO5
DO6
DO7
DO8
9
10
11
12
13
14
15
16
DATA LATCH
U12
74LS373
R39
3.3K
RP1
3.3K
+5V
VCC SUPPLY DAC
U5
7524
U6
R13
4.99K
1%
1/2W
VCC SUPPLY
741
U3
+24V
-9V
C7.1
C6
.1
Q2
VN46AF
R20
1K
R12
12.1K
R9
5K
RP2
3.3K
CE SUPPLY DAC
U7
7524
U6
R22
4.99K
1%
1/2W
CE SUPPLY
1436
U9
+48V
C13,.01/100V
Q3
MJE240
CR9
R31
R42
32.4K 1%
C14
27PF/200V
5K
R29
2.4K
1W
BIT SUPPLY DAC
U4
7524
U6
R11
4.99K
1%
1/2W
BIT SUPPLY
1436
U2
C8,.01/100V
C4
.01/100V
Q1
MJE240
R10
32.4K 1%
C5
27PF/200V
R6
5K
U8
7524
U6
R21
4.99K
1%
1/2W
R24 10K
1/2W 1%
U10
741
C16
C11
C15
.01
100V
148
+PROG
22
C27,1
C3,12,19,.01/100V
+48V
25
C26,.01/100V
-9V
LL
C25,22/15V
+24V
Z
C24,1
+5V
21
Y
C10,.01/100V
C28,31,.01/100V
GND
23
26
DD
C29,22/15V
U14
74LS04
C6
U
C5
17
A1'
K
A2'
L
A3'
M
A4'
N
C1
P
U13
74LS138
G1
G2B
G2A
CONTROL REG
U11
74LS273
U15A
74C74
U15B
74C74
SET
CK
CL
RESET MEMORY
R40
33K
R28
5K
VR1
431
DAC REF
C9,22
.01
100V
R23
1K
1/2W
TP6
5VREF
R38
3.3K
Q4
MJE240
CURRENT SENSE INTEGRATOR
R43
1K
TP1
SCOPE SYNC
R33
2.2K
CR10
C23
47/10V
R34
10K
Q7
2N2484
POWER ON RESET
V-Ø2
32
RR
35
SEL B SEL A
PP
RESET
COM.



REVISIONS						
LTR	DESCRIPTION		DR	CHK	APPR'D	DATE
A	RELEASE		MP	7/16	GS	7/20/82
B	ECN #4785	3-83	SH	TW	GS	3/83

NOTES: UNLESS OTHERWISE SPECIFIED.

1. ALL RESISTORS ARE IN OHMS, 1/4W, 5%.
2. ALL CAPACITORS ARE IN MICROFARADS, 50V.
3. LAST REF DES USED.
   RP2, U15, C31, Q8, CR12, R43, TP6, VR1.
4. POWER AND GROUND.

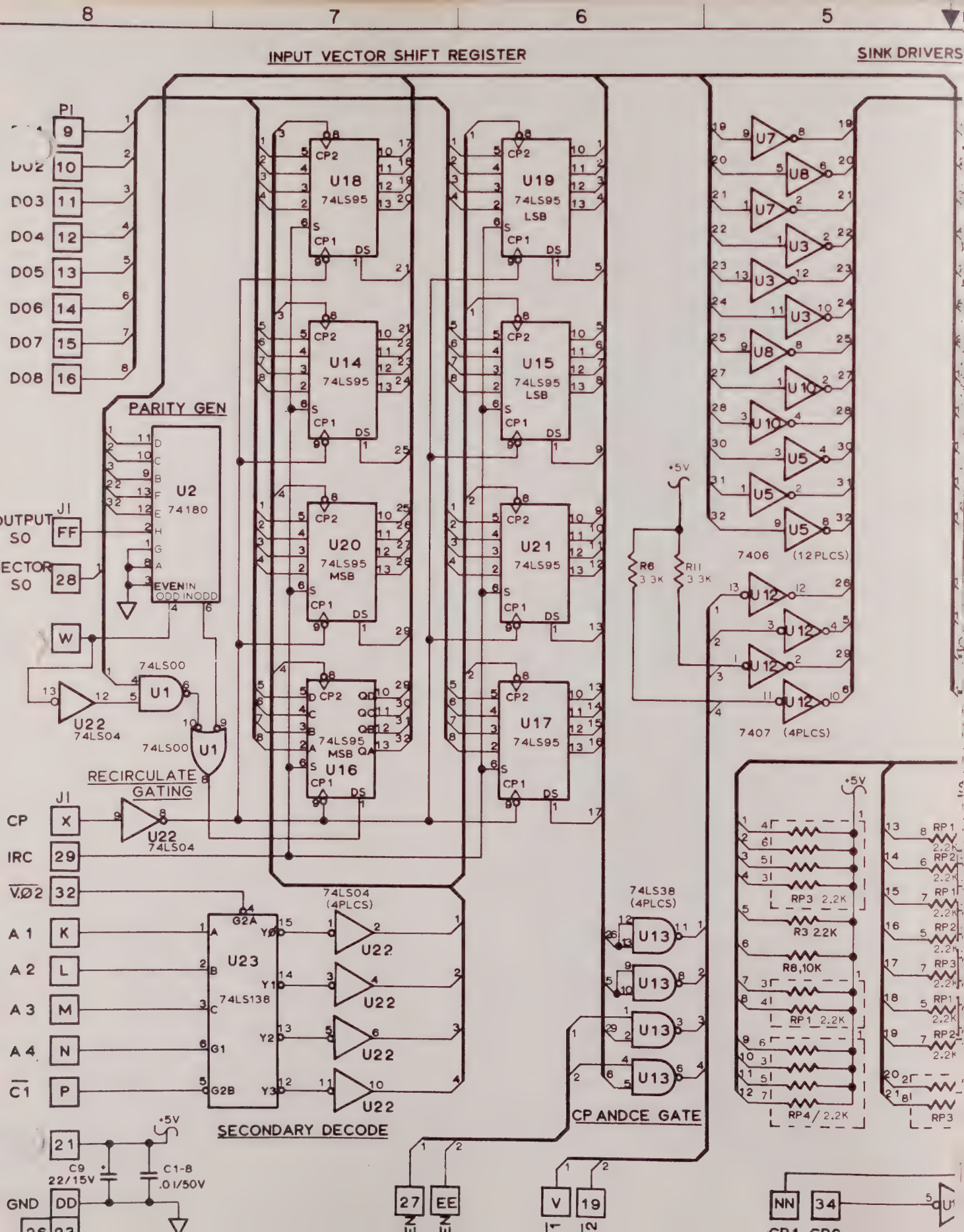
REF. DES.	+5V	GND	+24V	-9V
U14,15	14	7		
U11,12	20	10		
U13	16	8		
U6			4	11

5. ALL DIODES ARE 1N4148.

[6] -002 SHOWN.

701-1939-001 SHOWN, EXCEPT AS NOTED.

APPROVALS:		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES	DATA I/O	
DSN. ENGR.	DATE 4-4-83	TOLERANCES, UNLESS OTHERWISE SPECIFIED XX ± XXX ± ANGULAR	TITLE SCHEMATIC DIAGRAM, WAVE FORM GENERATOR	
MFG. ENGR.	DATE 7-7-82			
QUAL. ASSUR.	DATE 7/30/82	DRAWN BY: M Potter	DATE 8/12/82	SIZE D
ENGR. MGR.	DATE	CHECKED BY:	DATE	CODE IDENT NO 54193
				DRAWING NO 30-701-1939


8
7
6
5
INPUT VECTOR SHIFT REGISTER
SINK DRIVERS
PARITY GEN
U18
74LS95
U19
74LS95
LSB
U14
74LS95
U15
74LS95
LSB
U20
74LS95
MSB
U21
74LS95
U16
74LS95
MSB
U17
74LS95
U2
74180
U1
74LS00
U22
74LS04
7406 (12PLCS)
7407 (4PLCS)
R6 3.3K
R11 3.3K
+5V
RECIRCULATE GATING
SECONDARY DECODE
U23
74LS138
74LS04 (4PLCS)
74LS38 (4PLCS)
CP ANDCE GATE
RP1 2.2K
RP2 2.2K
RP3 2.2K
R3 2.2K
R8,10K
RP4/2.2K
C9 22/15V
C1-8 .01/50V
GND
P1
D02
D03
D04
D05
D06
D07
D08
OUTPUT SO
VECTOR SO
CP
IRC
V.Ø2
A1
A2
A3
A4
C1

REVISIONS					
LTR	DESCRIPTION	DR	CHK	APPR'D	DATE
A	RELEASE	S.H.			6-30
B	ECN #4785 3-83	SH			

J1

1 I/O 1 SINK/SOURCE
2 I/O 2 SINK/SOURCE
3 I/O 3 SINK/SOURCE
4 I/O 4 SINK/SOURCE
5 I/O 5 SINK/SOURCE, $\overline{CE2}$
6 I/O 6 SINK/SOURCE, CP2
7 I/O 7 SINK/SOURCE
8 I/O 8 SINK/SOURCE
12 I/O 9 SINK/SOURCE
11 I/O 10 SINK/SOURCE
10 I/O 11 SINK/SOURCE
9 I/O 12 SINK/SOURCE
15 I/O 13 SINK/SOURCE
14 I/O 14 SINK/SOURCE
13 I/O 15 SINK/SOURCE
16 I/O 16 SINK/SOURCE
46 I/O 17 SINK/SOURCE
45 I/O 18 SINK/SOURCE
44 I/O 19 SINK/SOURCE
43 I/O 20 SINK/SOURCE
50 I/O 21 SINK/SOURCE
49 I/O 22 SINK/SOURCE
48 I/O 23 SINK/SOURCE
47 I/O 24 SINK/SOURCE
29 I/O 25 SINK ONLY
28 I/O 26 SINK ONLY, $\overline{CE1}$
37 I/O 27 SINK/SOURCE
38 I/O 28 SINK/SOURCE
27 I/O 29 SINK ONLY, CP1
26 I/O 30 SINK ONLY
24 I/O 31 SINK ONLY
23 I/O 32 SINK ONLY

U8, U3, U5, U10, U7 — 7406 (16 PLCS)

RP1 2.2K, RP2 2.2K, RP4 2.2K, R5 2.2K, R10, 10K, +5V

7407 U12, RI2, R1

30 31 32 GND

NOTES: UNLESS OTHERWISE SPECIFIED

1. ALL RESISTORS ARE 2.2K OHMS, 1/4W, 5%.
2. ALL CAPACITORS ARE IN MICROFARADS.
3. LAST REFERENCE DESIGNATOR USED: U23, C9, RP4, RI2, J2
4.

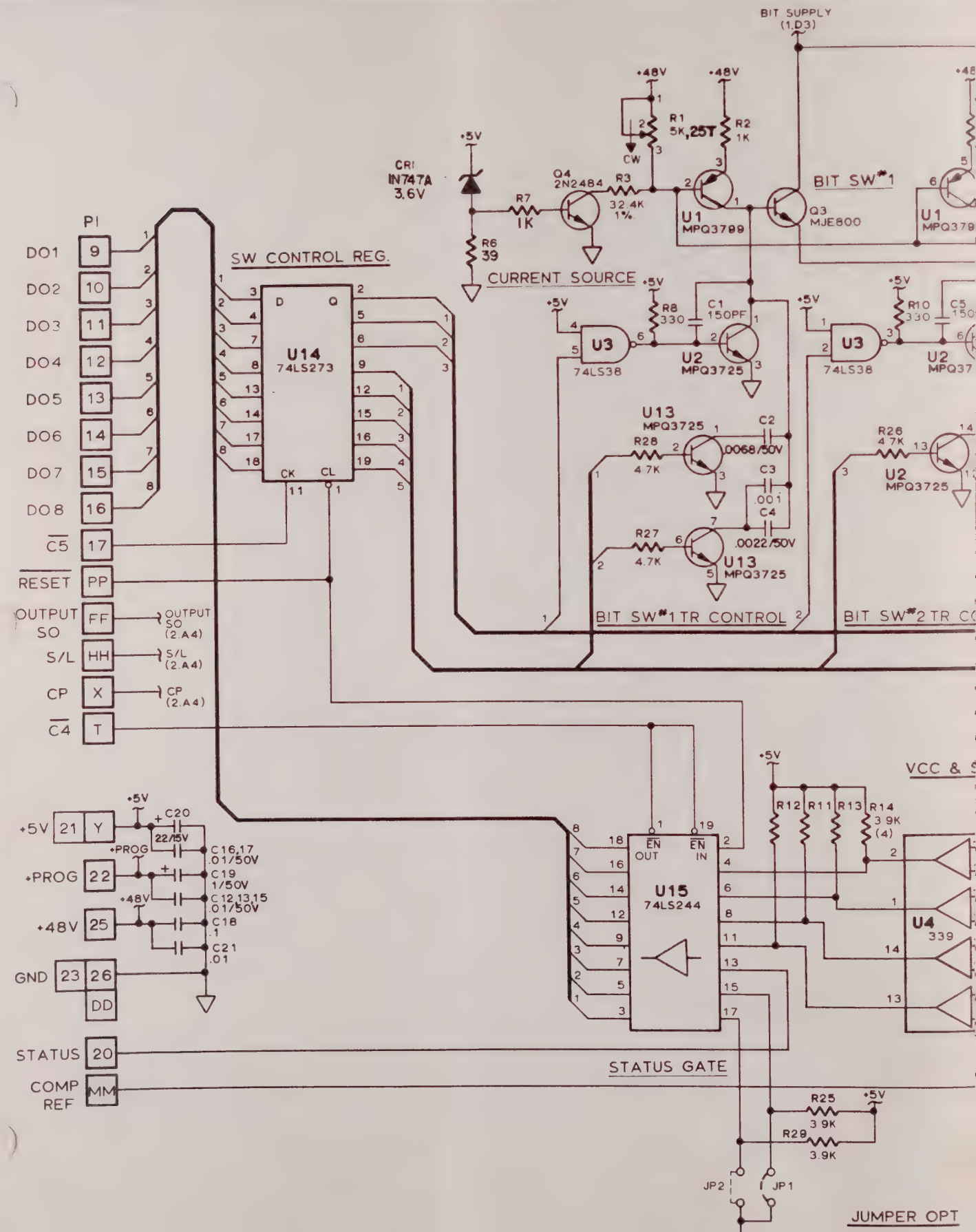
IC	+5V	GND
U1-3,5,7,8,10,12-22	14	7
U23	16	8

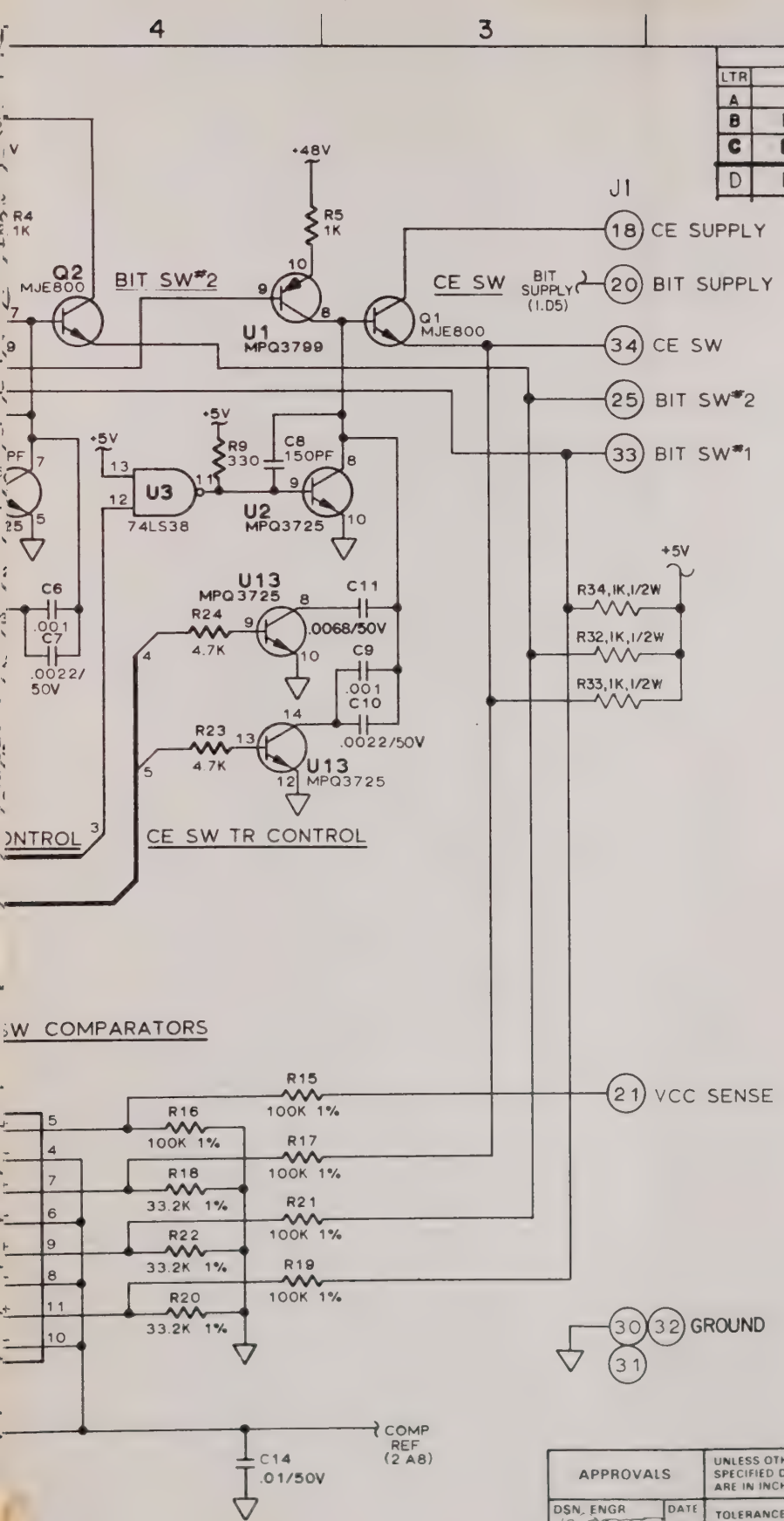
UNUSED GATES:

U3 7406 U3 7406

U1 74LS00 2 PLACES U1

APPROVALS		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES	DATA I/O			
DSN ENGR	DATE 11/11/82	TOLERANCES UNLESS OTHERWISE SPECIFIED XX ± XXX ± ANGULAR	TITLE SCHEMATIC DIAGRAM, SINK DRIVER			
MFG ENGR	DATE 11/12/82					
QUAL ASSUR	DATE	DRAWN BY JAEHEE KIM	DATE 12/81	SIZE D	CODE INDENT NO 54193	DRAWING NO 30-701-1940
ENGR MGR	DATE 11/18/82	CHECKED BY	DATE 1/11	SCALE		SHEET 1 OF 1


BIT SUPPLY (1,D3)
+48V
+48V
R1 5K,25T
R2 1K
CW
+5V
CR1 IN747A 3.6V
Q4 2N2484
R3 32.4K 1%
R7 1K
R6 39
U1 MPQ3799
Q3 MJE800
BIT SW#1
CURRENT SOURCE
P1
DO1
DO2
DO3
DO4
DO5
DO6
DO7
DO8
C5
RESET
OUTPUT SO
S/L
CP
C4
SW CONTROL REG.
U14 74LS273
D
Q
CK
CL
R8 330
C1 150PF
U3 74LS38
U2 MPQ3725
R10 330
C5
U13 MPQ3725
R28 4.7K
C2 .0068/50V
C3 .001
C4 .0022/50V
R27 4.7K
R26 4.7K
BIT SW#1 TR CONTROL
BIT SW#2 TR CO
OUTPUT SO (2.A4)
S/L (2.A4)
CP (2.A4)
VCC & S
+5V
+PROG
+48V
GND
C20 22/15V
C16,17 .01/50V
C19 1/50V
C12,13,15 .01/50V
C18 .1
C21 .01
U15 74LS244
EN OUT
EN IN
R12
R11
R13
R14 3.9K (4)
U4 339
STATUS
COMP REF
STATUS GATE
R25 3.9K
R29 3.9K
JP2
JP1
JUMPER OPT



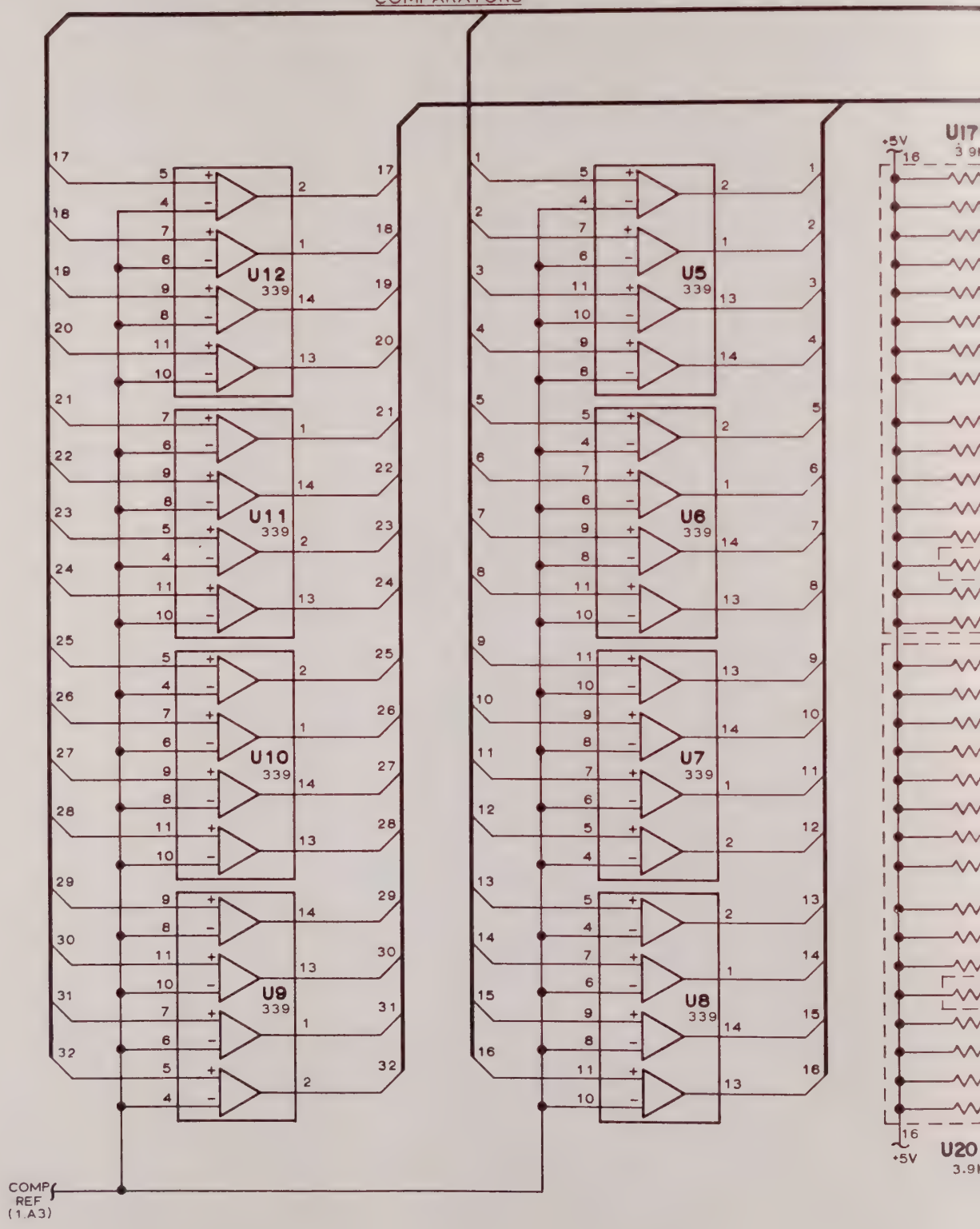
NOTES: UNLESS OTHERWISE SPECIFIED

1. ALL RESISTORS ARE IN OHMS 1/4W, 5%
2. ALL CAPACITORS ARE IN MFD, 100V
3. LAST REF DES USED
   R34, C21, JP2, Q7, CR1, U21, J2
4. POWER AND GROUND.

REF DES	+5	GND	+PROG
U3	14	7	
U4-U12		12	3
U14, 15	20	10	
U16,18,21,19	16	8	

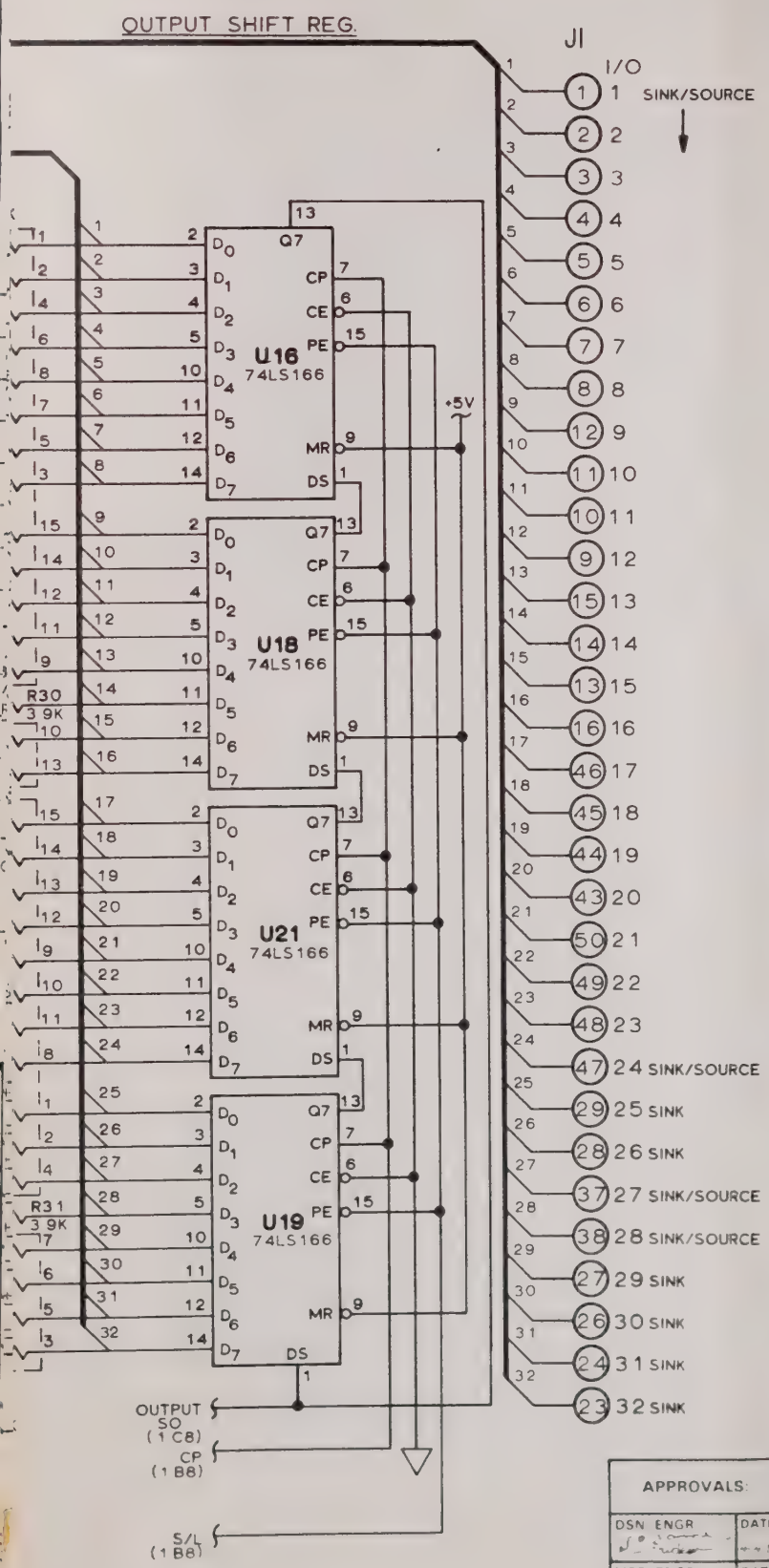
APPROVALS		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES	DATA I/O	
DSN. ENGR	DATE	TOLERANCES UNLESS OTHERWISE SPECIFIED XX ± XXX ± ANGULAR	TITLE SCHEMATIC DIAGRAM, T/RISE COMPARATOR	
MFG ENGR	DATE 7/26/82			
QUAL ASSUR	DATE	DRAWN BY	DATE 8/25/82	SIZE D / CODE IDENT NO 54193 / DRAWING NO 30-701-1941
ENGR MGR	DATE 11/15/82	CHECKED BY	DATE 4/1/82	SCALE — / SHEET 1OF2

COMPARATORS

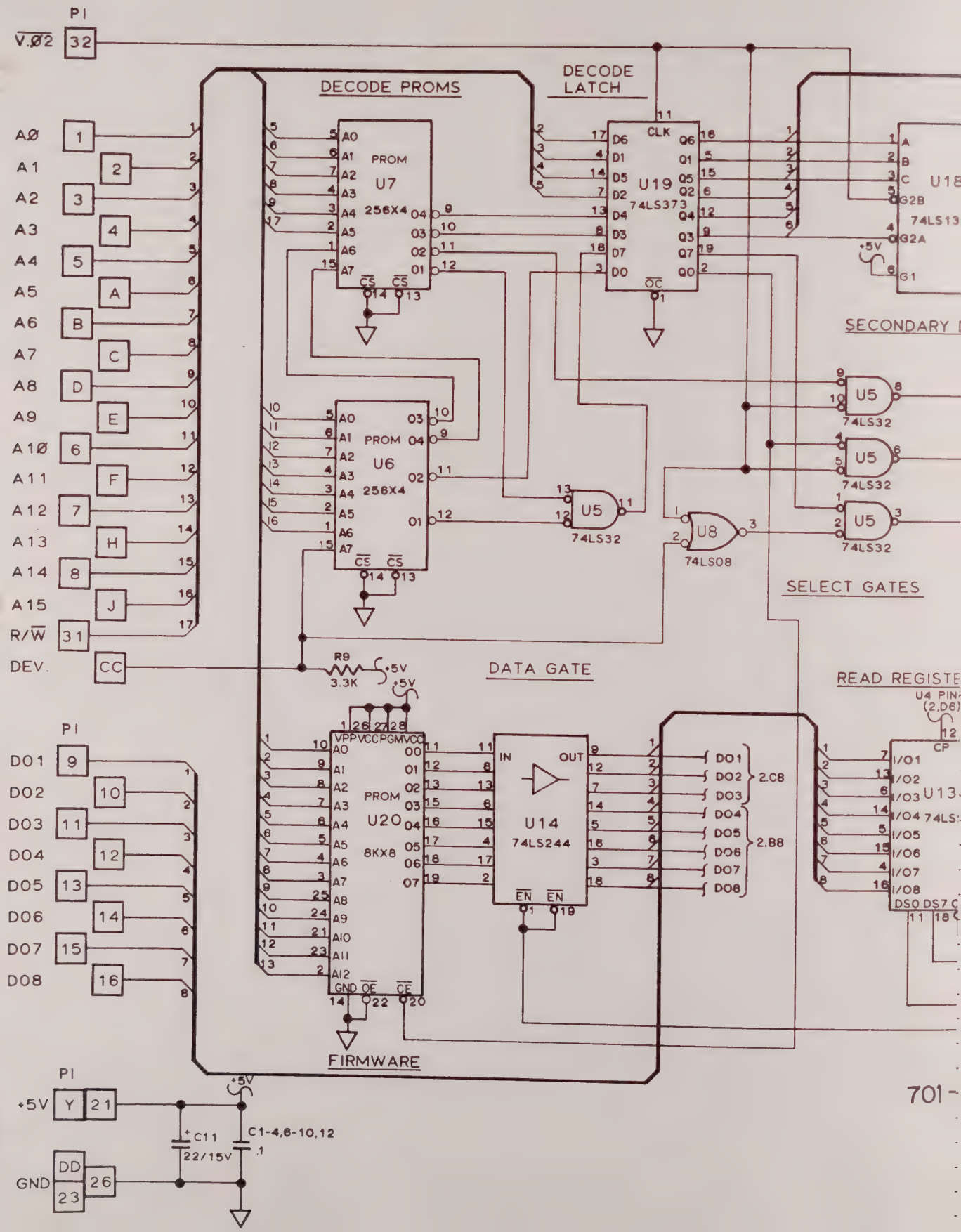


4 3 2 1

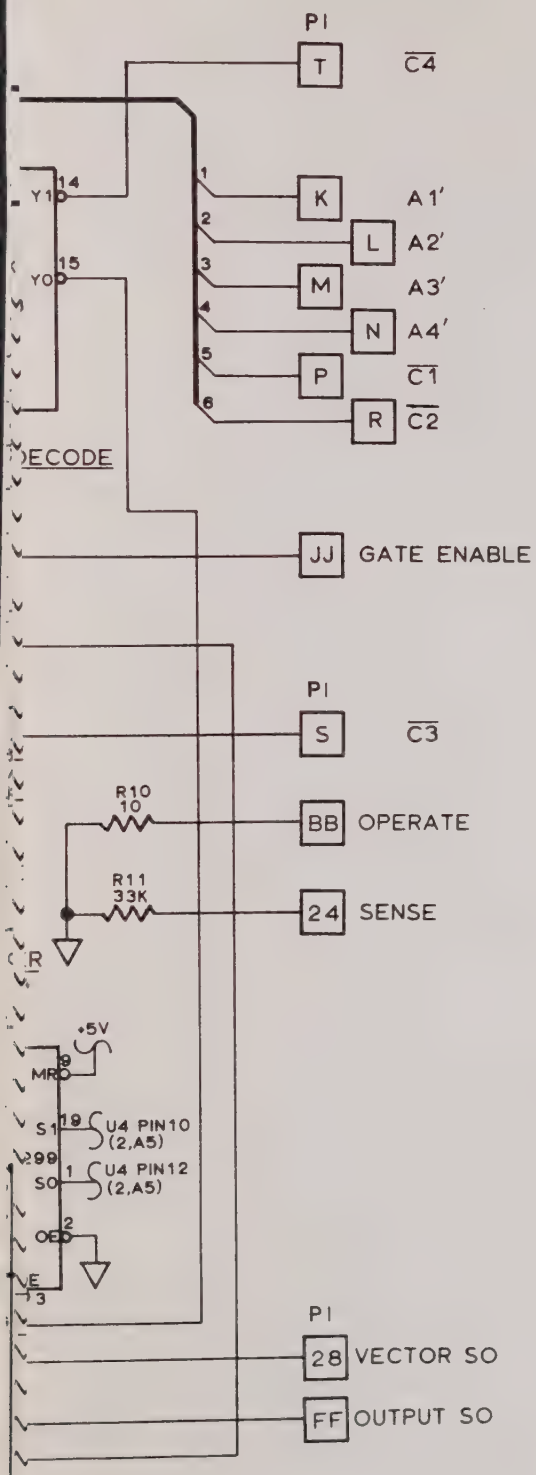
REVISIONS					
LTR	DESCRIPTION	DR	CHK	APPR'D	DATE
A	SEE SHT 1	MF		GS	6-30
B	SEE SHT 1	WK			10-20
C	SEE SHT 1				24-3
D	SEE SHT 1 3-83	SH	TW	GS	3/83



APPROVALS		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		DATA I/O			
DSN ENGR	DATE	TOLERANCES UNLESS OTHERWISE SPECIFIED XX ± XXX ± ANGULAR		TITLE SCHEMATIC DIAGRAM, T/RISE COMPARATOR			
MFG ENGR	DATE 2/4/82						
QUAL ASSUR	DATE 82	DRAWN BY M. Fatter	DATE 2/25/82	SIZE D	CODE IDENT NO 54193	DRAWING NO 30-701-1941	
ENGR MGR	DATE	CHECKED BY	DATE				


P1
V.Ø2
32
DECODE PROMS
DECODE LATCH
AØ
A1
A2
A3
A4
A5
A6
A7
A8
A9
A1Ø
A11
A12
A13
A14
A15
R/W
DEV.
CC
PROM
U7
256X4
U6
U19
74LS373
U5
74LS32
U8
74LS08
SECONDARY
SELECT GATES
DATA GATE
R9
3.3K
+5V
READ REGISTE
U4 PIN
(2,D6)
CP
U18
74LS13
G2B
G2A
G1
U13
74LS
DO1
DO2
DO3
DO4
DO5
DO6
DO7
DO8
2.C8
2.B8
PROM
U20
8KX8
U14
74LS244
IN
OUT
EN
FIRMWARE
P1
+5V
GND
C11
22/15V
C1-4,6-10,12
.1
701-

REVISIONS					
LTR	DESCRIPTION	DR	CHK	APPR'D	DATE
A	RELEASE	JK			1-30
B	ECN 4788	JK		69	3/83
C	ECN#4785 3-83	JK	TW	69	3/83



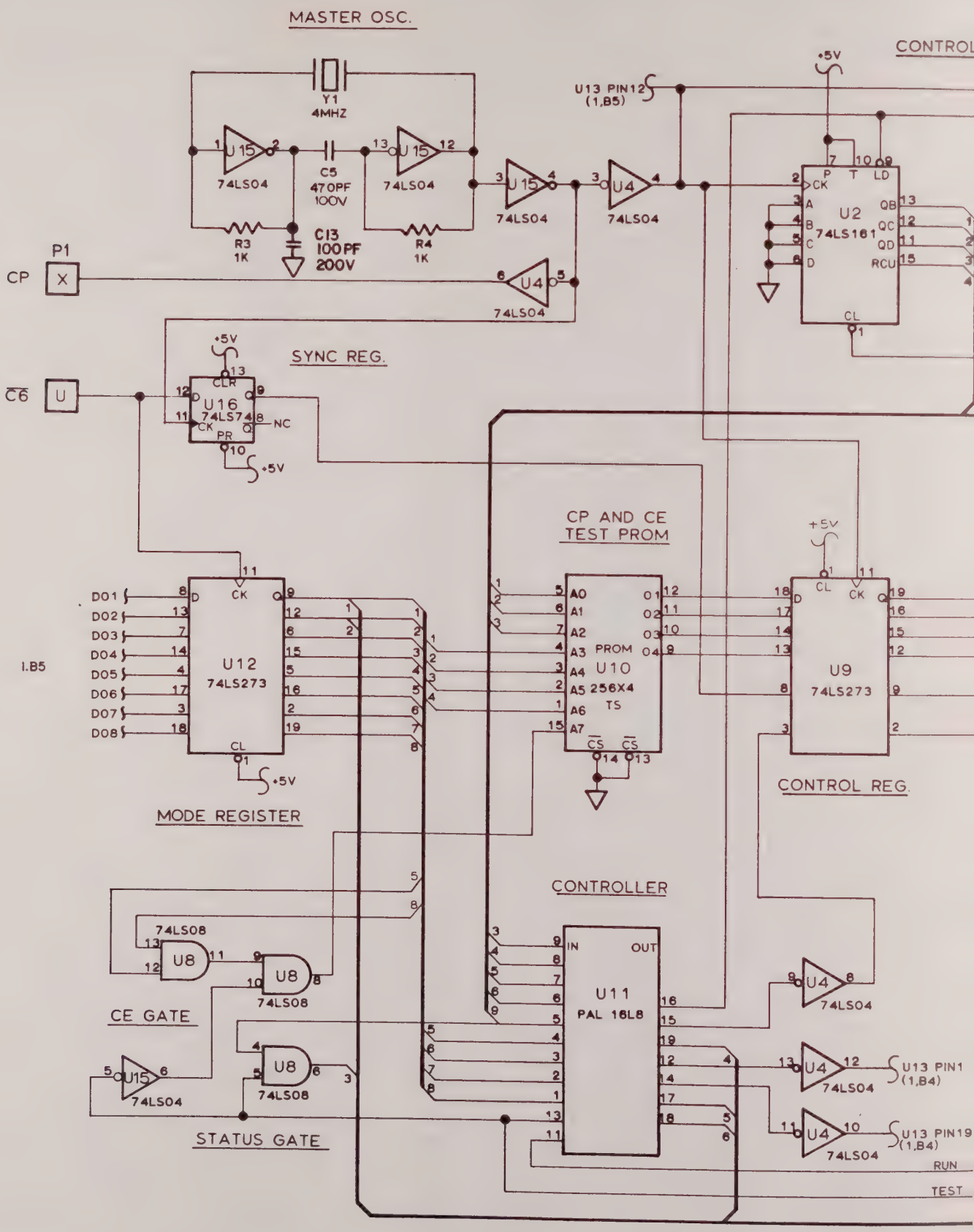
NOTES: UNLESS OTHERWISE SPECIFIED

1. ALL RESISTORS ARE IN OHMS, 1/4W, 5%.
2. ALL CAPACITORS ARE IN MICROFARADS, 50V.
3. LAST REFERENCE DESIGNATOR USED: U20, C13, R11, Y1, J1
4. POWER AND GROUND

REF DES	GND	+5V
4,5,8,15-17	7	14
U1,U2,3,6,7,10,18	8	18
U9 11,12,13,14,19	10	20

1942-002 SHOWN

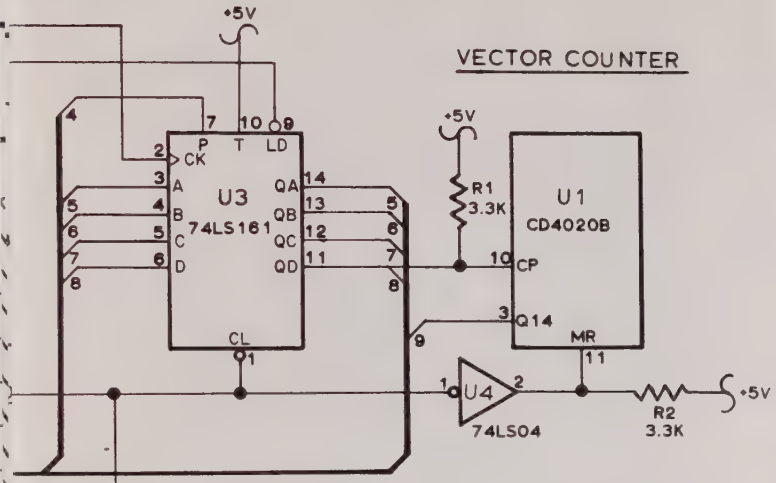
APPROVALS		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES	DATA I/O	
DSN ENGR	DATE	TOLERANCES, UNLESS OTHERWISE SPECIFIED XX ± XXX ± ANGULAR	TITLE SCHEMATIC DIAGRAM, CONTROLLER/MEMORY	
MFG ENGR	DATE			
QUAL ASSUR	DATE	DRAWN BY JAEHEE KIM	DATE 1/82	SIZE D
ENGR MGR	DATE	CHECKED BY	DATE 1/12	CODE IDENT NO 54193
				DRAWING NO 30-701-1942
				SCALE — SHEET 1OF2

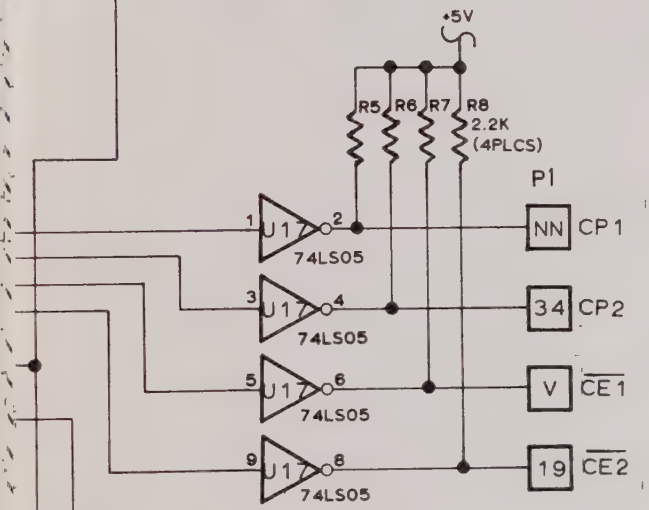

MASTER OSC.
CONTROL
Y1
4MHZ
U15
74LS04
C5
470PF
100V
C13
100PF
200V
R3
1K
R4
1K
U13 PIN12
(1,B5)
+5V
P1
CP
X
U4
74LS04
U2
74LS161
P
T
LD
CK
A
B
C
D
QB
QC
QD
RCU
CL
SYNC REG.
C6
U
U16
74LS74
CLR
CK
PR
NC
CP AND CE
TEST PROM
U10
PROM
256X4
TS
A0
A1
A2
A3
A4
A5
A6
A7
O1
O2
O3
O4
CS
U12
74LS273
D01
D02
D03
D04
D05
D06
D07
D08
MODE REGISTER
U9
74LS273
CONTROL REG.
CONTROLLER
U11
PAL 16L8
IN
OUT
74LS08
U8
CE GATE
STATUS GATE
U13 PIN1
(1,B4)
U13 PIN19
(1,B4)
RUN
TEST

REVISIONS					
LTR	DESCRIPTION	DR	CHK	APPR'D	DATE
A	SEE SHEET ONE	JK			6.30
B	"	JK		68	3/83
C	" 3-83	SH	TW	68	3/83

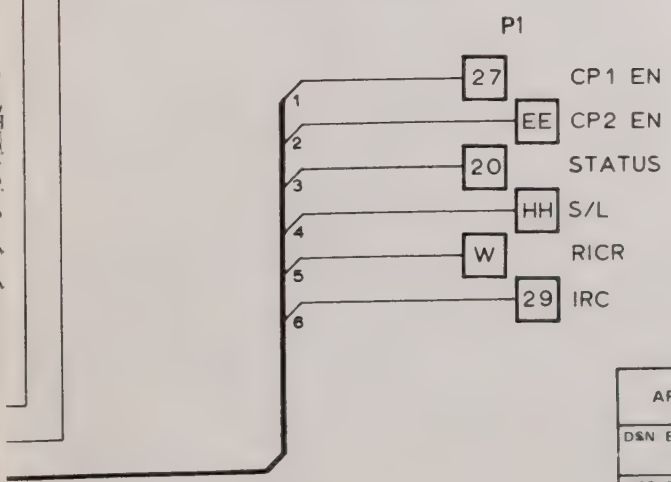
COUNTERS

VECTOR COUNTER

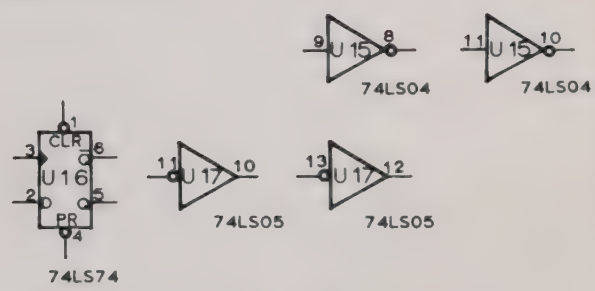




CP AND CE BUFFERS



UNUSED GATES:



APPROVALS		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		DATA I/O	
DSN ENGR	DATE	TOLERANCES UNLESS OTHERWISE SPECIFIED XX . XXX . ANGULAR		TITLE SCHEMATIC DIAGRAM, CONTROLLER/MEMORY	
MFG ENGR	DATE				
QUAL ASSUR	DATE	DRAWN BY JAEHEE KIM	DATE 1/82	SIZE D	CODE INDENT NO 54193
ENGR MGR	DATE	CHECKED BY	DATE	DRAWING NO 30-701-1942	
				SCALE	SHEET 2OF2

SOURCE DRIVERS

DATA LATCH

J3 & P1

DO1 29
DO2 27
DO3 25
DO4 23
DO5 21
DO6 19
DO7 17
DO8 15

U1 74LS373 (IN OUT G OC)

U7 74C374 (CLK D Q OC)

U6 74C374 (CLK D Q OC)

U4 74C374 (CLK D Q OC)

U5 74C374 (CLK D Q OC)

R6 3.3K

RP10 3.3K

J3 & P1

V·02 11
A1' 30
A2' 28
A3' 26
A4' 24
C1' 22

U2 74LS00

U3 74LS138 (A B C G1 G2A G2B Y0 Y1 Y2 Y3 Y4 Y5 Y6 Y7)

RP11 3.3K

SECONDARY DECODE

RESET 6

U2 74LS00

RP11 3.3K

+5 13 14 18

GND 1 2 9 10

C34, 35 100V

C33 22/15V

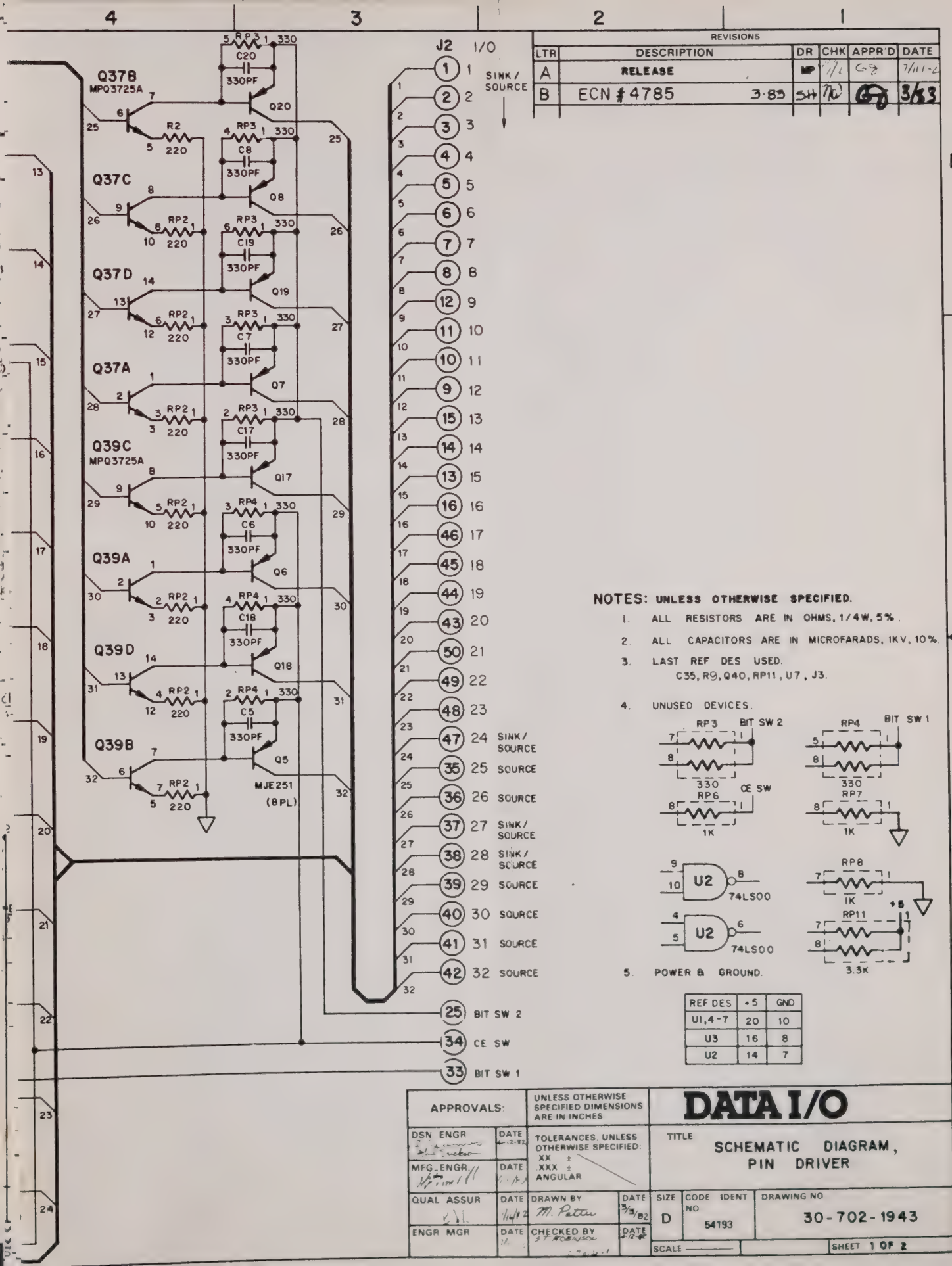
SOURCE REG

Q33C MPQ3725A
Q33D
Q33A
Q33B
Q34B MPQ3725A
Q34A
Q34D
Q34C
Q35B MPQ3725A
Q35A
Q35D
Q35C

R9 1K
C32 330PF Q32
RP8 1K
RP9 1K
C31 330PF Q31
C30 330PF Q30
C29 330PF Q29
C28 330PF Q28
R7 220
C27 330PF Q27
R8 220
C26 330PF Q26
C25 330PF Q25
RP6 1K
C23 330PF Q23
RP7 1K
C11 330PF Q11
C24 330PF Q24
C12 330PF Q12
MJE251 (12 PL)

Q36D MPQ3725A
Q36A
Q36C
Q36B
Q40B MPQ3725A
Q40A
Q40D
Q40C
Q38D MPQ3725A
Q38C
Q38A
Q38B

RP6 1K
C22 330PF Q22
RP7 1K
C10 330PF Q10
RP4 330
C21 330PF Q21
R5 220
C9 330PF Q9
R4 220
R3 330
C16 330PF Q16
RP1 220
RP5 330
C3 330PF Q3
R1 220
C15 330PF Q15
C4 330PF Q4
C14 330PF Q14
C1 330PF Q1
C2 330PF Q2
C13 330PF Q13
MJE251 (12 PL)


4
3
2
1
REVISIONS
LTR | DESCRIPTION | DR | CHK | APPR'D | DATE
A | RELEASE | MP | | | 7/11-2
B | ECN # 4785 3-83 | SH | | | 3/83
J2 I/O
SINK/ SOURCE
Q37B MPQ3725A
Q37C
Q37D
Q37A
Q39C MPQ3725A
Q39A
Q39D
Q39B
MJE251 (8PL)
RP3 330
C20 330PF Q20
R2 220
C8 330PF Q8
RP2 220
C19 330PF Q19
C7 330PF Q7
C17 330PF Q17
RP4 330
C6 330PF Q6
C18 330PF Q18
C5 330PF Q5
25 BIT SW 2
34 CE SW
33 BIT SW 1
NOTES: UNLESS OTHERWISE SPECIFIED.
1. ALL RESISTORS ARE IN OHMS, 1/4W, 5%.
2. ALL CAPACITORS ARE IN MICROFARADS, 1KV, 10%.
3. LAST REF DES USED.
C35, R9, Q40, RP11, U7, J3.
4. UNUSED DEVICES.
RP3 BIT SW 2
RP4 BIT SW 1
330
CE SW
RP6
RP7
1K
U2
74LS00
RP8
RP11
+5
3.3K
5. POWER & GROUND.
REF DES | +5 | GND
U1,4-7 | 20 | 10
U3 | 16 | 8
U2 | 14 | 7
APPROVALS:
UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES
TOLERANCES, UNLESS OTHERWISE SPECIFIED:
XX ±
XXX ±
ANGULAR
DSN ENGR
MFG. ENGR.
QUAL ASSUR
ENGR MGR
DRAWN BY M. Patten
CHECKED BY S. T. Robinson
DATE 3/3/82
DATA I/O
TITLE
SCHEMATIC DIAGRAM, PIN DRIVER
SIZE D
CODE IDENT NO 54193
DRAWING NO 30-702-1943
SCALE
SHEET 1 OF 2

J3 & P1
(50 PIN)
SOME PINS USED
ON SHEET 1
J4
(96 PIN)
47 A0 88
48 A5 85
45 A1 82
46 A6 79
43 A2 76
44 A7 73
39 A3 67
40 A8 66
37 A4 64
38 A9 68
35 A10 61
36 A11 65
33 A12 58
34 A13 62
31 A14 55
32 A15 59
29 DO1 52
20 C3 47
27 DO2 49
12 DEV 29
25 DO3 46
3 SEL A 14
23 DO4 43
4 SEL B 13
21 DO5 40
8 GATE ENABLE 19
19 DO6 37
5 R/W 17
17 DO7 34
7 IRQ 20
15 DO8 31
13 14 16 18 +5 1 2 3 28
6 RESET 16
1 2 9 10 GND 22 23 26
11 V·02 25

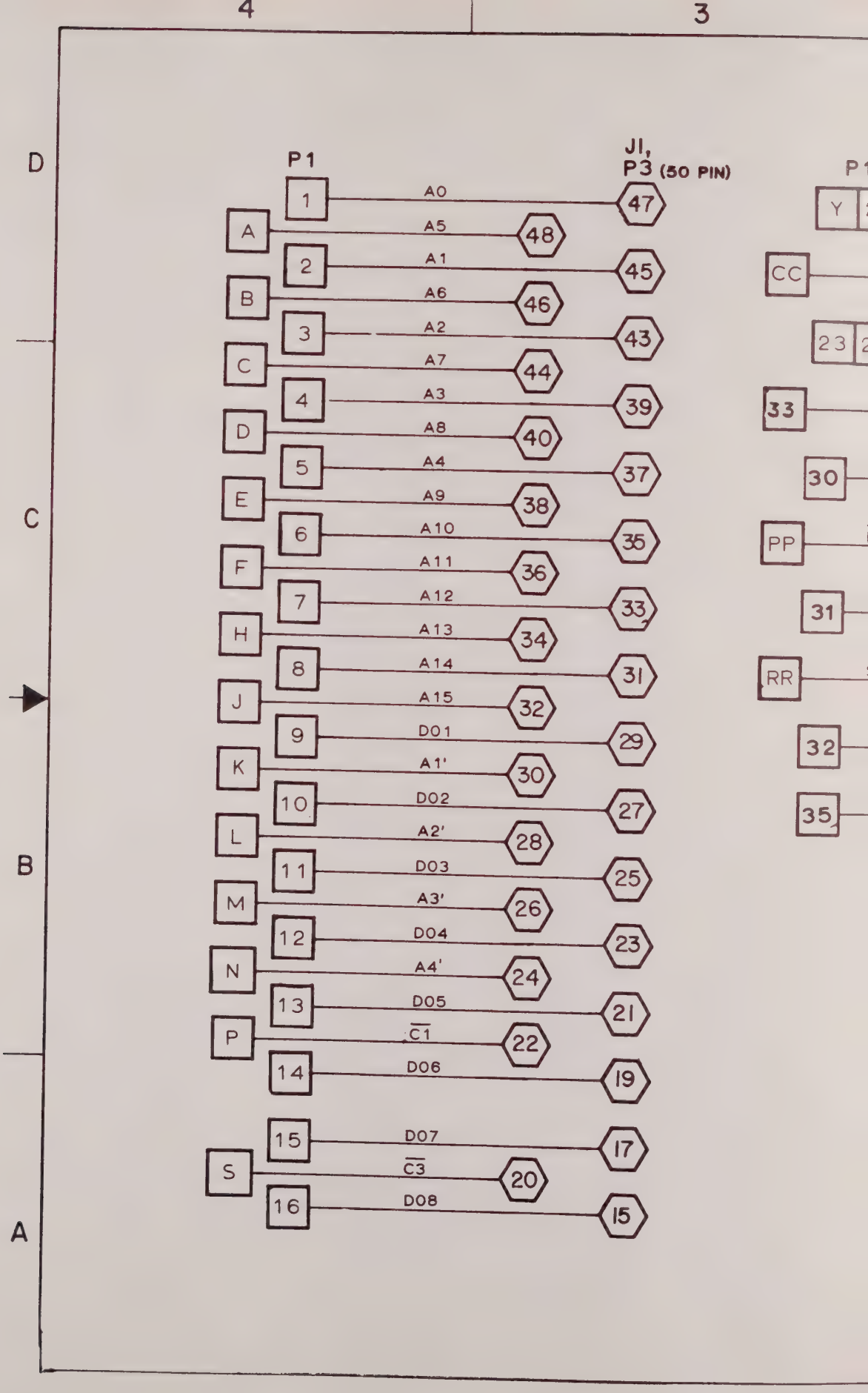
J2
(50 PIN)
SOME PINS USED
ON SHEET 1
1 2 3 4 5 6 7 8 12 11 10 9 15 14 13 16 46 45 44 43 50 49 48 47 29 35 28 36 37 38 27 39 26 40 24 41 23 42 22 21 20 18 19 17 25 33 34 30 31 32

REVISIONS					
LTR	DESCRIPTION	DR	CHK	APPR'D	DATE
A	SEE SHT 1				
B	" 3-83	SH	MW	GB	3/83

J4
(96PIN)

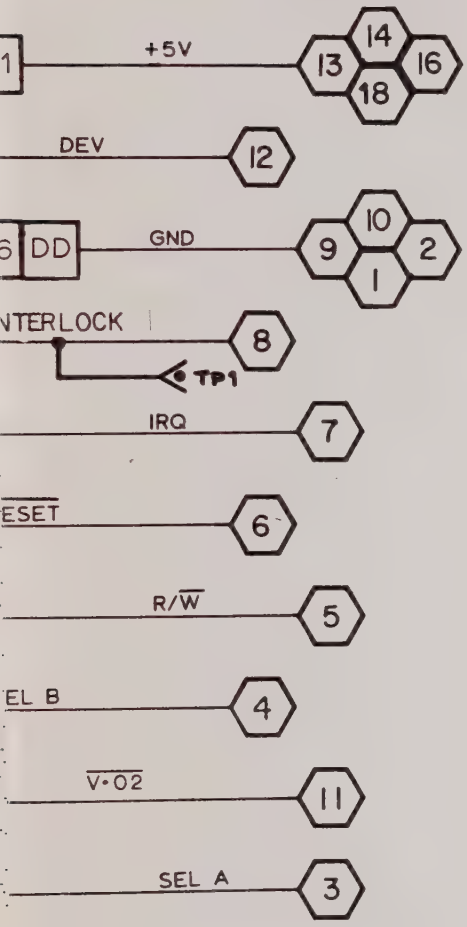
I/O 1	15
I/O 2	18
I/O 3	21
I/O 4	24
I/O 5	27
I/O 6	30
I/O 7	32
I/O 8	33
I/O 9	39
I/O 10	38
I/O 11	36
I/O 12	35
I/O 13	44
I/O 14	42
I/O 15	41
I/O 16	45
I/O 17	90
I/O 18	89
I/O 19	87
I/O 20	86
I/O 21	96
I/O 22	95
I/O 23	93
I/O 24	92
I/O 25	69
I/O 25	74
I/O 26	63
I/O 26	75
I/O 27	77
I/O 28	78
I/O 29	60
I/O 29	80
I/O 30	56
I/O 30	81
I/O 31	53
I/O 31	83
I/O 32	54
I/O 32	84
VCC	6
VCC SENSE	51
BIT SUPPLY	9
CE SUPPLY	12
VFE CLAMP	50
VPUP	48
BIT SW 2	57
BIT SW 1	91
CE SW	94
GND	70, 71, 72

APPROVALS:		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		DATA I/O			
DSN. ENGR. G. Jamison John Tucker	DATE 4-12-83	TOLERANCES UNLESS OTHERWISE SPECIFIED .XX ± .XXX ± ANGULAR		TITLE SCHEMATIC DIAGRAM, PIN DRIVER			
MFG ENGR.	DATE						
QUAL ASSUR	DATE	DRAWN BY M. Patten	DATE 5/3/2	SIZE D	CODE IDENT NO 54193	DRAWING NO 30-702-1943	
ENGR. MGR	DATE	CHECKED BY: J. T. Robison	DATE 4-12-83	SCALE		SHEET 2 OF 2	


4
3
D
C
B
A
P1
J1,
P3 (50 PIN)
1
A
2
B
3
C
4
D
5
E
6
F
7
H
8
J
9
K
10
L
11
M
12
N
13
P
14
15
S
16
A0
A5
A1
A6
A2
A7
A3
A8
A4
A9
A10
A11
A12
A13
A14
A15
D01
A1'
D02
A2'
D03
A3'
D04
A4'
D05
C1
D06
D07
C3
D08
47
48
45
46
43
44
39
40
37
38
35
36
33
34
31
32
29
30
27
28
25
26
23
24
21
22
19
17
20
15
P1
Y
CC
23
33
30
PP
31
RR
32
35

2 1

REVISIONS					
LTR	DESCRIPTION	DR	CHK	APPR'D	DATE
A	RELEASE	R.B.			11/1/82
B	ECN #4685	CL		GT	2/17/83
C	ECN #4785	SH	MW	GT	3/83



NOTES:

1. LAST REFERENCE DESIGNATOR USED: P3, J1, TP1

APPROVALS:		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES	DATA I/O	
DSN ENGR	DATE	TOLERANCES UNLESS OTHERWISE SPECIFIED XX ± XXX ± ANGULAR	TITLE SCHEMATIC DIAGRAM, EXTENDER BOARD 50/72	
MFG ENGR	DATE			
QUAL ASSUR	DATE	DRAWN BY R. BENN	DATE 12-17-81	SIZE C / CODE IDENT NO 54193 / DRAWING NO 30-701-1944
ENGR MGR	DATE	CHECKED BY	DATE	SHEET 1 OF 1

U.S. and Canada Sales Offices and Representatives
Direct Sales and Service Offices

HAWAII, OREGON, WASHINGTON

Data I/O Corporation
10525 Willows Road N.E.
C-46
Redmond, WA 98052
(206) 881-6444
Telex 15-2167
(Sales & Service)

NORTHERN CALIFORNIA

Data I/O Corporation
473 Sapena Court
Suite 4
Santa Clara, CA 95050
(408) 727-0641
Telex 352054
(Sales, Applications & Service)

WESTERN REGION

Data I/O Corporation
2770 South Harbor Blvd.
Suite K
Santa Ana, CA 92704
(714) 662-1182
Telex 910-595-1562
(Sales & Service)

EASTERN REGION ADMINISTRATIVE OFFICES

MAINE, EASTERN MASSACHUSETTS, NEW HAMPSHIRE, VERMONT

Data I/O Corporation
Nashua Road
Route 101A
Amherst, NH 03031
(603) 889-8511 *(Sales)*
(603) 889-8513 *(Applications & Service)*
Telex 943431

SOUTHERN REGION ADMINISTRATIVE OFFICE

Data I/O Corporation
1810 N. Glenville Drive
Suite 108
Richardson, TX 75081
(214) 235-0044
Telex 792474
(Sales & Applications)

Representatives

(Sales Only)

ALASKA

Microprocessor Technology
3537 East 65th Avenue
Anchorage, AK 99507
(907) 344-8080
Telex (090) 25-462

OREGON, EASTERN WASHINGTON

Northwest Test and Measurement
8196 Southwest Hall
Suite 217
Beaverton, OR 97005
(503) 646-9966

ARKANSAS, LOUISIANA, OKLAHOMA, TEXAS

TESTECH, Inc.
1000 E. Campbell Road
Suite 108
Richardson, TX 75081
(214) 644-5010

9219 Katy Freeway
Suite 124
Houston, TX 77024
(713) 467-2192

2525 Wallingwood
Suite 804
Austin, TX 78746
(512) 327-7033

MICHIGAN, EASTERN KENTUCKY, OHIO, WEST VIRGINIA, WESTERN PENNSYLVANIA

Electro Sales Associates
1635 Mardon Drive
Dayton, OH 45432
(513) 426-5551

29200 Vassar Road
Suite 505
Livonia, MI 48152
(313) 474-7320

9816 Portage Road
Portage, MI 49002
(616) 323-2416

851 East 222nd Street
Cleveland, OH 44123
(216) 261-5440

3740 Mount Royal Blvd.
Allison Park, PA 15101
(412) 487-3801

UPSTATE NEW YORK

Martin P. Andrews
523 East Genesee Street
P.O. Box 443
Fayetteville, NY 13066
(315) 637-5291

METROPOLITAN NEW YORK, LONG ISLAND, NORTHERN NEW JERSEY

RTI
300 Northern Blvd.
Great Neck, NY 11021
(516) 829-3770

KANSAS, MISSOURI, NEBRASKA, SOUTHERN ILLINOIS

Midtec Associates, Inc.
7702 Mize Road
DeSoto, KS 66018
(913) 441-6565
Telex 910-743-4129

8460 Lindbergh North
Suite 11
Florissant, MO 63031
(314) 837-5200

ALABAMA, FLORIDA, GEORGIA, MISSISSIPPI, NORTH CAROLINA, SOUTH CAROLINA, TENNESSEE

Pen Tech Associates
1398 Semoran Blvd.
Suite 106
Casselberry, FL 32707
(305) 645-3444

201 S.E. 15th Terrace
Suite K
Deerfield Beach, FL 33441
(305) 421-4989

Cherokee Center
627 Cherokee Street
Suite 21
Marietta, GA 30060
(404) 424-1931

Holiday Office Center
3322 Memorial Pkwy. S.W.
Suite 36
Huntsville, AL 35801
(205) 881-9298

3709 Alliance Drive
Suite C
Greensboro, NC 27407
(919) 852-6000

INDIANA, IOWA, WESTERN KENTUCKY, MINNESOTA, NORTHERN ILLINOIS, NORTH DAKOTA, SOUTH DAKOTA, WISCONSIN

Torkelson Associates
4940 Viking Drive
Minneapolis, MN 55435
(612) 835-2414

108 Wilmot Road
Suite 110
Dearfield, IL 60015
(312) 945-8700

2346 S. Lynhurst
Room B101
Indianapolis, IN 46241
(317) 244-7867

2840 N. Brookfield Road
Brookfield, WI 53005
(414) 784-7736

ARIZONA, COLORADO, IDAHO, MONTANA, NEW MEXICO, NEVADA, UTAH, WYOMING

Zeus Electronics, Inc.
1428 E. Pierson Street
Phoenix, AZ 85014
(602) 263-6022
Telex 910-951-1362

310 Val Verde S.E.
Albuquerque, NM 87108
(505) 842-6633

3333 Quebec Street
Suite 2950
Denver, CO 80207
(303) 321-4246

1849 W. North Temple
Suite B105
Salt Lake City, UT 84116
(801) 534-0500

SOUTHERN NEW JERSEY, DELAWARE, MARYLAND, EASTERN PENNSYLVANIA, VIRGINIA, WASHINGTON D.C.

SCI-REP, Inc.
304 Cooper Center
Pennsauken, NJ 08109
(609) 662-5222

9512A Lee Highway
Fairfax, VA 22031
(703) 385-0600

CONNECTICUT, RHODE ISLAND, WESTERN MASSACHUSETTS

Pat Jenks Associates
52 Washington Avenue
North Haven, CT 06473
(203) 239-6201

114 Cummings Park
Woburn, MA 01801
(617) 938-0488

CANADA

Allan Crawford Associates, Ltd.
6503 Northam Drive
Mississauga, Ontario L4V 1J2
(416) 678-1500
Telex 06 968769

881 Lady Ellen Place
Ottawa, Ontario K1Z 5L3
(613) 722-7682
Telex 053 3600

7018 Cote De Liesse
St. Laurent, P.Q. H4T 1E7
(514) 731-8564
Telex 05-824944

1935-30th Avenue N.E.
Calgary, Alberta T2E 6Z5
(403) 230-1341
Telex 03 821186

3795 William Street
Burnaby, B.C. V5C 3H3
(604) 294-1326
Telex 04 54247

192 Joseph Zatzman Drive
Dartmouth, Nova Scotia B3B 1N4
(902) 463-9360
Telex 019 31604

15043A 118th Avenue
Edmonton, Alberta T5V 1H9
(403) 451-4893

Addresses on this list subject to change without notice.

0321/183

International Sales Offices and Representatives

Europe

Data I/O Europe
Vondelstraat 50-52
NL-1054 GE Amsterdam
The Netherlands
(20)186855
Telex 16616 DATIO NL

Data I/O Germany GmbH
Bahnhofstrasse 3
D-6453 Seligenstadt
West Germany
(6182)3088/89
Telex 4184962 DATA D

AUSTRIA

Ing. Ernst Steiner
Hummelgasse 14
A-1130 Wien
Austria
(222)827474
Telex 135026 ES A

BELGIUM

Simac Electronics
Rue du Progres 52
BOITE 3
B-1000 Brussels
Belgium
(2)2192451-3
Telex 23662 SIMEIP B

DENMARK

ITT Komponent A/S
Fabriksparken 31
DK-2600 Glostrup
Denmark
(2)451822
Telex 33355 itt dk

FINLAND

Havulinna-Instrumentarium Oy
P.O. Box 357
SF-00101 Helsinki 10
Finland
(0)799711
Telex 124426 havul sf

FRANCE

M.B. Electronique
606, rue Fourny
Zac de Buc
P.O. Box 31
F-78530, Buc, France
(3)9568131
Telex MB 695414 F

GERMANY

Instrumatic Electronics GmbH
Am Kirchenholzel 14
D-8032 Grafelfing (near Munich)
West Germany
(89)852063
Telex 524298 INSTR D

GREECE

Mr. Elias Mammeas
65 Meg. Alexandrou & Athinas Street
P.O. Box 181
Korydallows
Piraeus
(1)4967815/or 818
Telex 213835 LH GR

ISRAEL

R.D.T. Electronics Engineering, Ltd.
46 Sokolov Street
P.O. Box 75
Ramat Hasharon
Israel
(3)483211
Telex 32143 RDT IL

ITALY

Sistrel SPA
Via Giuseppe Armellini 39
I-00143 Roma
Italy
(6)5915551
Telex 680356 SISTREL i

Sistrel SPA
Via P. da Volpedo 59
I-20092 Cinisello Balsamo (Mi)
Italy
(2)6120129 or 6181893
Telex 334643 SISTREL i

NETHERLANDS

Simac Electronics
Veenstraat 20
NL-5503HR Veldhoven
The Netherlands
(40)533725
Telex 51037 SIMAC NL

NORWAY

Teleinstrument A/S
P.O. Box 134
N-1371 Asker
Norway
(2)789460
Telex 72919 TELIN N

PORTUGAL

Decada
Equipamentos de Electronica, Lda.
Rua Pedro Nunes 47-C
P-1000 Lisboa or P-1003 Lisboa Codex
Portugal
(19)574984
Telex 18469 NONIO P

SPAIN

Instrumatic
Alameda Principal 26
Apartado 151
Malaga 5
Spain
(52)213199/213898
Telex 77131 Hafn e

Instrumatic
Juan Hurtado de Mendoza 9
Madrid-16
Spain
(1)2502577/2507278
Telex 46277 Itic e

SWEDEN

Macrotek AB
Vallingbyvagen 212, Box 43
S-162 11 Vallingbyvagen, Sweden
(8)870190
Telex 12543 MATEK S

SWITZERLAND

Instrumatic SA
5-7, rue du Clos
CH-1207 Geneve
Switzerland
(22)360830
Telex 28667 FLUJA CH

Instrumatic AG
Weingartenstrasse 9
CH-8803 Ruschlikon
Switzerland
(1)7241410
Telex 56605 INST CH

TURKEY

Data I/O Europe
Vondelstraat 50-52
NL-1054 GE Amsterdam
The Netherlands
(20)186855
Telex 16616 DATIO NL

UNITED KINGDOM

Microsystem Services
P.O. Box 37
Lincoln Road
Cressex Industrial Estate
High Wycombe
Bucks, HP12 3XJ, England
(494)41661
Telex 837187 MICSYS G

International

Data I/O International
10525 Willows Road N.E.
C-46
Redmond, WA 98052
U.S.A.
(206) 881-6444
Telex 15-2167

AUSTRALIA

CORPORATE OFFICE

Warburton O'Donnell Ltd.
372 Eastern Valley Road
P.O. Box 182
Chatswood, N.S.W., Australia 2067
4073261
Telex Warfran AA21299

SOUTH AUSTRALIA

ADELAIDE, AUSTRALIA

Warburton Franki (Adelaide) Pty. Ltd.
322 Grange Road
Kidman Park
South Australia 5025 Australia
3567333
Telex Warfran AA82579

BRISBANE, QUEENSLAND

Warburton Franki (Brisbane) Pty. Ltd.
13 Chester Street
Fortitude Valley
Queensland 4006 Australia
527255
Telex Warfran AA41052

MELBOURNE, VICTORIA

Warburton Franki (Melbourne) Pty. Ltd.
220 Park Street
South Melbourne
Victoria 3205 Australia
699 4999
Telex Warfran AA31370

PERTH, WESTERN AUSTRALIA

Warburton Franki (Perth)
98-102 Belgravia Street
Belmont, 6104
Western Australia
65 7000
Telex Warfran AA92908

SYDNEY, NEW SOUTH WALES

Warburton Franki Sydney
199 Parramatta Road
Auburn, N.S.W. 2144
Australia
648 1711
Telex Warfran AA22265

TASMANIA

Associated Agencies Pty., Ltd.
P.O. Box 252
43 Albert Road
Moonah, Tasmania 7009
23 1841
Telex ASSAG AA58206

BRAZIL

Cosele
Instrumentos Electronicos Ltda.
Rue da Consolacao, 867
3.° And. - Conj. 32
01301 - Sao Paulo - SP
Brasil
255-1733 or 256-7421 or 231-5548
Telex (011) 30869 CSEL-BR

HONG KONG

Cummings Engineering Co., Ltd.
90 Sung Wong Toi Road, I/F,
To Kwa Wan, Kowloon,
Hong Kong
3-636227
Telex 74015 CUMMH HX

INDIA

Transmarketing Private, Ltd.
Sterling Center
16/2 Dr. Annie Besant Rd.
Bombay 400-18
India
022 391874
Telex 011-5424

JAPAN

Data I/O Japan Company, Ltd.
Sankei Building
1-8, Sarugaku-cho 2-Chome
Chiyoda-Ku, Tokyo 101
Japan
(03) 295-2656
Telex 2225391 DATAIO J

KOREA

Elcom System, Inc.
Yeongdong Box 50
Seoul, Korea
555-5222 or 557-3836
Telex ADUCEL K25227

MALAYSIA

Cadco Technical Services
67A, Jalan Perisai
Taman Sri Tebrau
Johore Bahru, Malaysia
Cable - CADCOKIT - JOHOR BAHRU

MEXICO

Christensen, S.A.
Guillermo Prieto 76-304
Col. San Rafael
Delegacion Cuahutemoc
06470-Mexico, D.F.
546-25-95
546-29-55
Telex 017-75612 Mycome.

NEW ZEALAND

Wellington-New Zealand:
Warburton Franki, Ltd.
42-43 Oxford Terrace
Lower Hutt, New Zealand
693-016
Telex Warfran NZ 3824

Warburton Franki, Ltd.
P.O. Box 9301; Newmarket
142 Broadway
Auckland, New Zealand
504-458
Telex Warfran NZ 3824

PHILIPPINES

Cummings Engineering Co., Ltd.
90 Sung Wong Toi Road, 1/F,
To Kwa Wan, Kowloon
Hong Kong
3-636227
Telex 74015 CUMMH HX

SINGAPORE

GEA Technology PTE., Ltd.
Units 1003 to 1008, Block 3, 10th Floor
PSA Multi-Storey Complex
Pasir Panjang Road
Singapore 0511
2729412
Telex RS 37162 Answerback Geasin

SOUTH AFRICA

Electronic Building Elements
(PTY) Ltd.
P.O. Box 4609
Pine Square, 18th Street
Hazelwood
Pretoria, South Africa 0001
46-9221/7
Telex 3 0181 SA
Telegrams Elbilem

SOUTH & CENTRAL AMERICA

Dexter Computer Products, Inc.
3465 Torrance Blvd.
Suite N
Torrence, CA 90503
213-542-7333
Telex 182360 DXTR TRNC

TAIWAN

COMPAC Microelectronics, Inc.
11th Fl., Walsin Bldg.
#219, Chung-Hsiao E. Road, Sec. 4
P.O. Box 53-479
Taipei, Taiwan R.O.C.
7529911
Telex 24368 COMTWN

THAILAND

Dynamic Supply Engineering R.O.P.
12 Soi Pasara 1, Sukhumvit 63
Bangkok-11, Thailand
Tel. - 3914434, 39228532
Telex 82455 DYNASUP TH

Warranty 90 Days

Data I/O warrants that all equipment sold pursuant to any resultant agreement shall be free from defects in material or workmanship at the time of delivery. Such warranty shall extend for 90 days. Buyer must provide notice to Data I/O within this prescribed warranty period of any defect; if the defect is not the result of improper usage, service, maintenance or installation and equipment has not been otherwise damaged or modified after delivery, Data I/O shall either replace or repair the defective part or parts of equipment or replace the equipment or refund the purchase price at Data I/O's option after return of such equipment by buyer to Data I/O. Shipment to and from Data I/O's facility shall be borne on account of buyer.

(a) Consequential Damages — Data I/O shall not be liable for any incidental or consequential damages incurred as a result of any defect in any equipment sold hereunder, and Data I/O's liability is specifically limited to its obligation described herein to repair or replace a defective part or parts covered by this warranty.

(b) Exclusive Warranty — The warranty set forth herein is the only warranty, oral or written, made by Data I/O and is in lieu of and replaces all other warranties, expressed or implied, including the WARRANTY OF MERCHANTABILITY AND THE WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE.

0299/6K/183

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 41 REDMOND, WA 98052

POSTAGE WILL BE PAID BY ADDRESSEE

DATA I/O
Reader Comments
Applications Manager
10525 Willows Road N.E.
C-46
Redmond, WA 98052

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 41 REDMOND, WA 98052

POSTAGE WILL BE PAID BY ADDRESSEE

DATA I/O
Warranty/Registration
Service Manager
10525 Willows Road N.E.
C-46
Redmond, WA 98052

Reader Comments

The manual's completeness, accuracy, organization, usability, and reliability: _____

_____

Did you find errors in this manual? _____

How can this manual be improved? _____

_____

_____

Additional comments: _____

_____

Name _____ Title _____

Department _____ M/S _____

Company _____

Address _____

City _____ State _____ Zip _____

Phone ( ) _____

Manual or Part No. _____

Warranty/Registration

Please return this card to Data I/O within five days

Did the packaging of this equipment exhibit any outward signs of physical damage? . ☐ YES ☐ NO

Did this equipment arrive intact, without loose parts or cable damage? ..... ☐ YES ☐ NO

Did the equipment operate on power-up? ..... ☐ YES ☐ NO

Did you attain adequate system performance? ..... ☐ YES ☐ NO

Were any electrical adjustments required? ..... ☐ YES ☐ NO

If you required assistance, was a local Data I/O representative contacted? ..... ☐ YES ☐ NO

Comments: _____

_____

Name _____ Title _____

Department _____ M/S _____

Company _____

Address _____

City _____ State _____ Zip _____

Phone ( ) _____

Model or Description _____ Serial or Part No. _____

Date Received _____

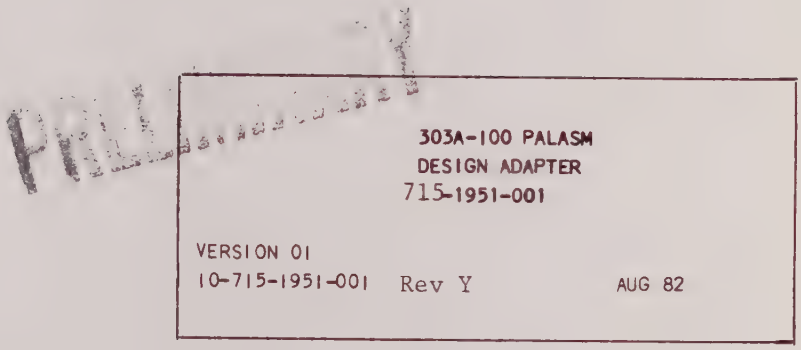
In order to deliver your PLDS at the earliest possible date, preliminary manuals are being supplied with several elements of the PLDS. They will provide you with all the information required to operate the PLDS, but do not include circuit descriptions, or troubleshooting and repair information.

The final Rev A manuals and binders will be sent to your shipping address automatically.

If you have questions about the equipment that are not explained in the preliminary manual, contact your nearest Data I/O Service Center. A Field Applications Engineer will be happy to answer any of your questions. Telephone numbers are provided at the end of the manual.

Data I/O Corporation

/cjh


303A-100 PALASM
DESIGN ADAPTER
715-1951-001
VERSION 01
10-715-1951-001 Rev Y
AUG 82

Data I/O has made every attempt to ensure that the information in this document is accurate and complete. However, Data I/O assumes no liability for an damages that result from use of this document or the equipment which it accompanies.

Data I/O reserves the right to make changes to this document without notice at any time.

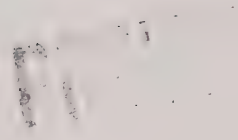


TABLE OF CONTENTS

SECTION 1. INTRODUCTION

1.1 GENERAL INFORMATION
1.2 APPLICATIONS
1.3 FIELD APPLICATIONS SUPPORT
1.4 WARRANTY
1.5 SERVICE
1.6 ORDERING

SECTION 2. INSTALLATION AND MAINTENANCE

2.1 INSPECTION
2.2 INSTALLATION

SECTION 3. DATA DEVELOPMENT

3.1 INTRODUCTION
3.2 OVERVIEW
3.3 PREOPERATIONAL PROCEDURES
3.4 THE FUNCTION MENU
3.5 DATA DEVELOPMENT
- 3.5.1 Receive Source Equation (2)(E2)
- 3.5.2 Receive Fuse Data (B)(EB)
- 3.5.3 Edit Source Equations (a)
- 3.5.4 Translate Source Equations (4)(E4)
- 3.5.5 Simulate Source Equations (5)(E0)
- 3.5.6 Display Fuse Pattern (A)(EA)
- 3.5.7 Enter Decimal Fuse Data (E)(EE)
- 3.5.8 Enter Functional Test Data (8)(E8)(E9)
- 3.5.9 Verify Option (6)(E6)
- 3.5.10 Enter Last Fuse Option (7)(E7)
- 3.5.11 Transmit Source Equations (3)(E3)
- 3.5.12 Transmit Fuse Data (C)(EC)

3.6 PALASM ERROR CODES

LIST OF FIGURES

1-1 303A-100 PALASM Adapter
2-1 303A-100 PALASM Adapter
3-1 Flowchart, Data Development
3-2 Source Equations Example
3-3 Function Menu
3-4 Receive Source Equations
3-5 Receive Fuse Data
3-6 Edit Menu
3-7 Enter Source Equations Example
3-8 Flowchart, Edit Source Equations
3-9 Simulate Source Equations
3-10 Logic Diagram
3-11 Fuse Pattern

LIST OF TABLES

2-1 Required Equipment
3-1 PALASM Menu Commands
3-2 The Editing Command Set

SECTION 1. INTRODUCTION

1.1 GENERAL INFORMATION

The 303A-100 PALASM (fig 1-1) design adapter consists of interface circuitry and EPROM memory, mounted in a metal frame. It is a software development aid used with the Data I/O LogicPak to enable you to define logic structure and reduce it to a fuse pattern. The 303A-100 PALASM Design Adapter is a subsystem of the Programmable Logic Development System (PLDS). The PLDS was created to develop data, program, verify, and test currently available logic devices.

The information in this manual is divided into three parts. The introduction section gives you general information on the design adapter and specifics on field applications support, warranty, service, and ordering. The installation section tells how to install the adapter. The data development section describes the operations used to develop and reduce equations. Read the manual thoroughly before attempting to develop data with the PALASM Design Adapter.

1.2 APPLICATIONS

The design adapter helps you define the logic structure that you intend to program. It is usually used in conjunction with a terminal connected to the programmer. Its firmware translates your inputs, including part numbers and logic descriptors, into a fuse pattern and testing format. The resultant logic structure is contained in the programmer RAM. This enables the design adapter to be removed from the LogicPak, the correct programming/testing (p/t) adapter to be installed, and the fuse pattern programmed directly into the device.

Alternately, the logic structure can be offloaded at any development stage for extended storage via (1) the serial port to disk, tape or printer or (2) PROM programming pak to PROM.

When the design adapter is installed on the LogicPak, normal Load, Program, and Verify operations are disabled.

The Data I/O Programmable Logic Applications Chart in the back of this manual lists all the logic devices currently available, those compatible with the 303A-100 PALASM Design Adapter, as well as the necessary PLDS components used to develop, program, verify, and test each. When new devices become available, the chart will be updated.

1.3 FIELD APPLICATIONS SUPPORT

Data I/O Field Application Engineers (FAE's) throughout the world can provide additional information about interfacing Data I/O products with other equipment, and answer questions about your equipment.

The locations of the FAE's within the United States are given on the address list at the back of this manual. For international applications support, contact your nearest Data I/O representative.

1.4 WARRANTY

Data I/O equipment is warranted against defects in materials and workmanship. The warranty period of ninety days begins when you receive the equipment.

The warranty card inside the back cover of this manual explains the length and conditions of the warranty. For warranty service, contact your nearest Data I/O Service Center.

1.5 SERVICE

Data I/O maintains service centers throughout the world staffed with factory-trained technician to provide prompt, quality service. A list of all service centers is located at the back of this manual.

1.6 ORDERING

To place an order for equipment, contact your Data I/O sales representative. Orders for shipment must include the following information:

- Description of the equipment. (See the latest Data I/O Price List or contact your sales representative for equipment and part numbers.)
- Purchase order number.
- Desired method of shipment.
- Quantity of each item ordered.
- Shipping and billing address of the firm, including zip code.
- Name of person ordering equipment.

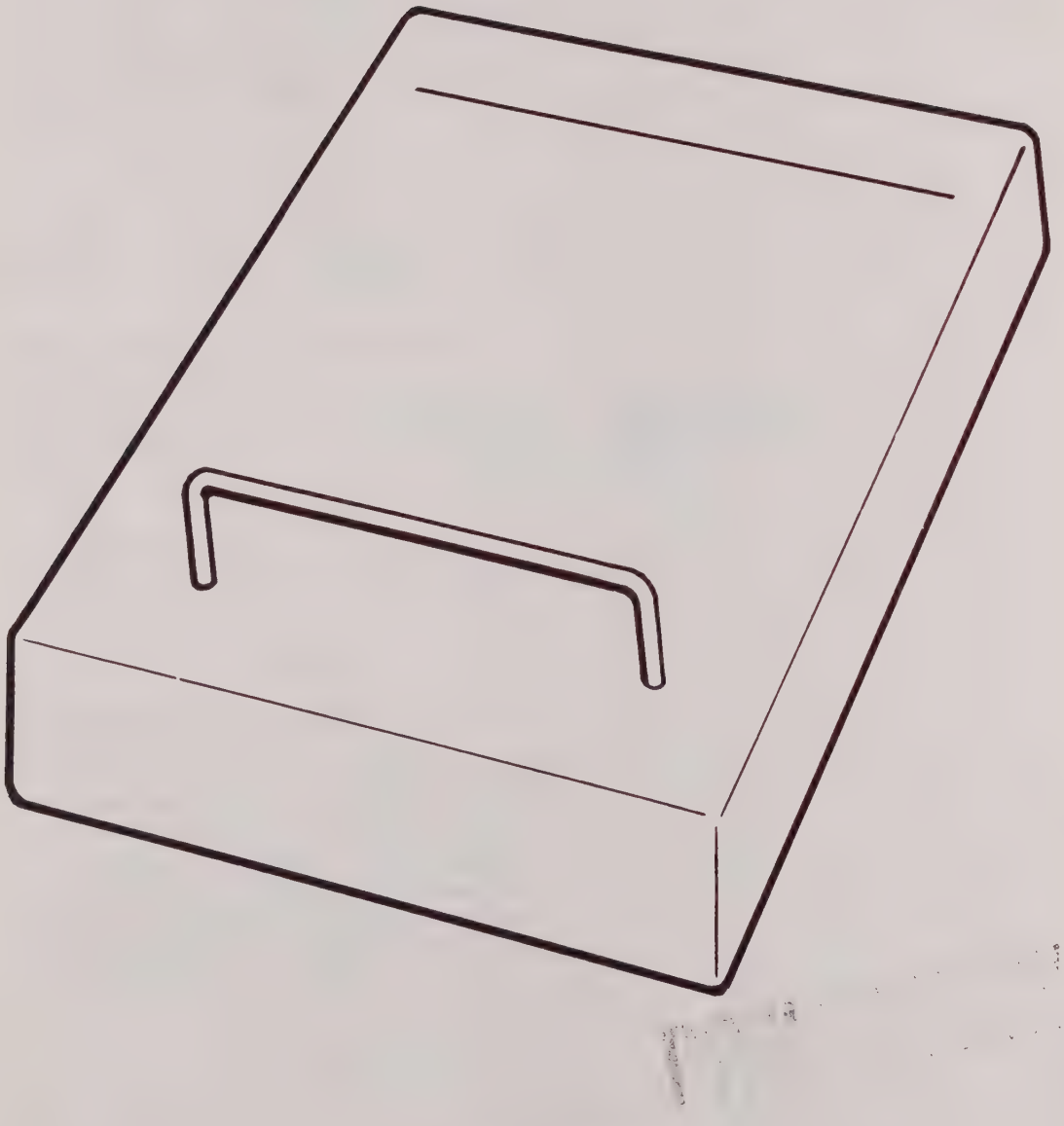


Figure 1-1. 303A-100 PALASM Adapter

SECTION 2. INSTALLATION AND MAINTENANCE

2.1 INSPECTION

The PALASM Design Adapter was thoroughly tested and inspected before shipment. For trouble-free operation, verify on receipt that it is in the best possible condition. All the equipment listed in table 2-1 should be present. Inspect the adapter for any damage that may have occurred during transit. If you note any damage, file a claim with the carrier and notify Data I/O. Read the information in sections 2.2 and 2.3 before using the design adapter for data development.

2-1. Required Equipment

DESCRIPTION	DATA I/O PART NUMBER	QTY
303A-100 PALASM Design Adapter	324-1951-001	1
Instruction Manual	10-324-1951-001	1

2.2 INSTALLATION

Install the design adapter in the LogicPak as shown in figure 2-1.

CAUTION

BEFORE REMOVING THE ADAPTER, enter an <esc> from the terminal; or from the programmer front panel press the KEYBOARD key (on the System 19) or the VERIFY key (on the 100A or 29A). The Processor in the programmer executes firmware resident in the adapter. These precautions must be taken before removing the adapter from the LogicPak to prevent a program interrupt and resulting loss of RAM data.

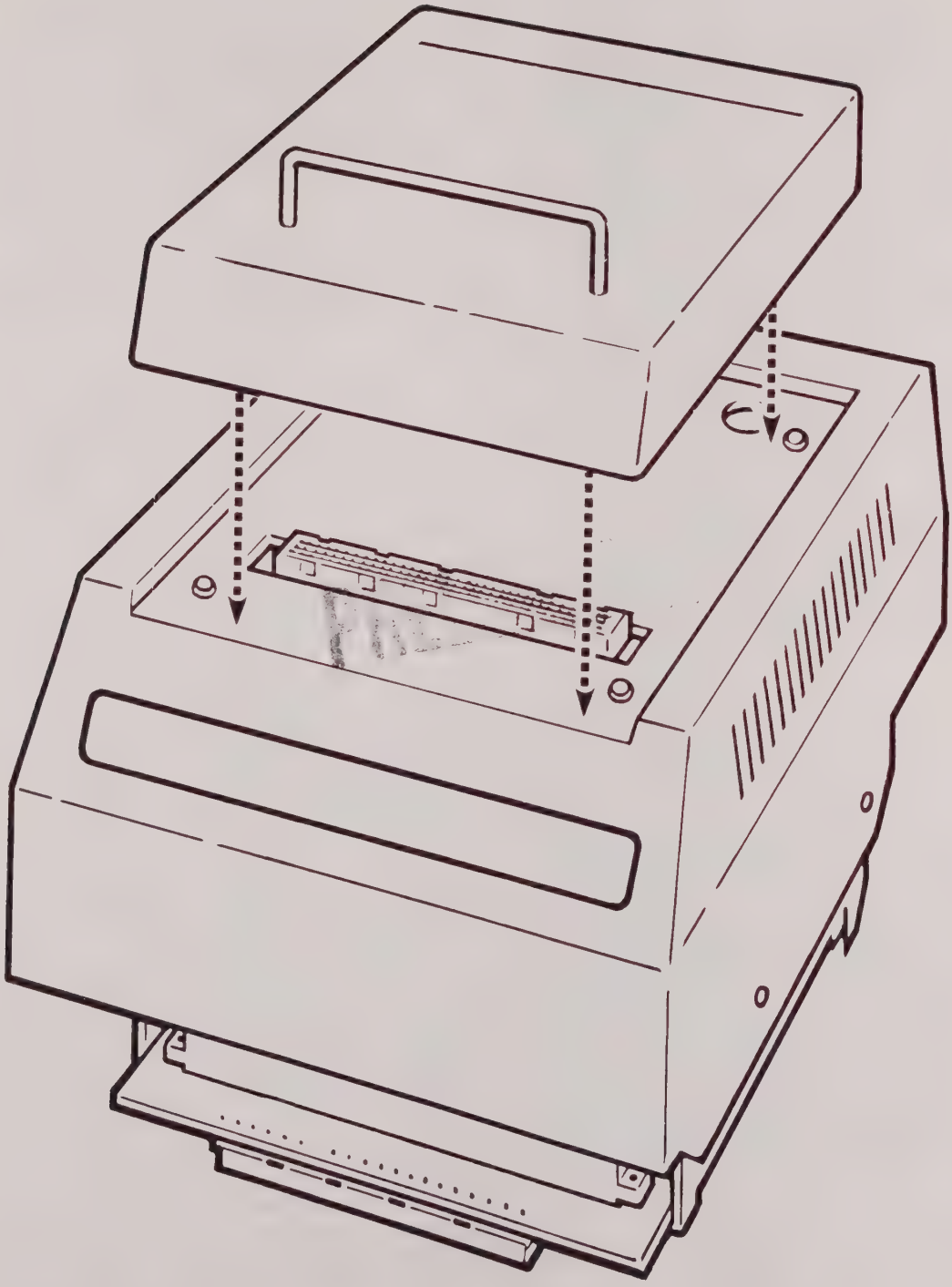


Figure 2-1. 303A-100 PALASM Design Adapter

SECTION 3. DATA DEVELOPMENT

3.1 INTRODUCTION

This section explains the steps necessary for developing data using the PALASM design adapter. An overview of the PALASM data development process is given in section 3.2. Section 3.3 gives the necessary preoperational procedures and section 3.4 outlines the commands and functions used. Section 3.5 describes the data development process.

We recommend that you familiarize yourself with the commands, functions, and overall flow of events before starting to develop data with the PALASM Design Adapter.

3.2 OVERVIEW

Figure 3-1 is a flowchart of the data development process. Data development with the 303A-100 PALASM Design Adapter is best accomplished with a PLDS configuration consisting of the LogicPak, design adapter, terminal, and a Data I/O programmer. The terminal, though not essential for some operations, is required for entering design information and all other (reperture) operations in the design adapter's repertoire that are not performed by the p/t adapter.

Data development begins with source equations and function tables (truth tables). They describe the function you want the device to perform. Figure 3-2 shows Boolean equations for the MMI PAL 12H6, along with conventional logic symbology and PAL logic symbology. The 12H6 is used to illustrate data development features of the PALASM design adapter throughout this manual.

The design adapter simplifies the tedious step of translating source equations into a fuse pattern--a series of 0's and 1's that signify which fuses need to be programmed in order for the device to perform its desired function.

Design adapter firmware allows you to specify a device and enter source equations and function tables (1) manually from the terminal keyboard or (2) from other sources via the RS-232 serial port in the programmer.

When commanded, firmware (1) compares equations to function table to test the validity of your inputs, (2) generates test vectors if they are valid, (3) translates the equation information into a fuse pattern, and (4) downloads the fuse pattern to the programmer RAM. Editing can be performed at any stage, at equation or fuse levels. The development adapter also contains firmware for extensive test and verify options.

After data has been downloaded to RAM, the design adapter can be removed from the LogicPak and replaced with a p/t adapter. The device is then programmed with the fuse states existing in the programmer RAM.

For more information on data development, programming, and verifying and testing of logic device, consult the LogicPak manual.

3.3 PREOPERATIONAL PROCEDURES

The following steps must be performed to ready the PLDS for data development with the PALASM design adapter:

1. Install the design adapter as described in Section 2 of this manual.
2. Set the parity, stop bits, and baud rate so they are the same on all the equipment.
3. Set terminal for:
   a) Disable any special functions using CNTL H, CNTL M, CNTL P, CNTL Z.
   b) Select distructive cursor for proper display during edit mode.
4. Power-up the programmer and terminal.
5. Press SELECT.
6. Enter E1.
7. Press START.
8. Enter the Family and Pinout Codes from the terminal.
9. The function menu will be displayed.

The menu explanations in section 3.5 are intended to supplement the more complete discussion of these functions and commands in the LogicPak manual. Refer to the LogicPak manual for additional information when necessary.

Most commands can be entered with a single keystroke from the terminal, and the sequence requires no explanation. Options that require entry of additional data are described in text as they would be performed from a terminal. Data entry from the front panel is outlined in the LogicPak Manual and fully described in text when it differs from terminal operation.

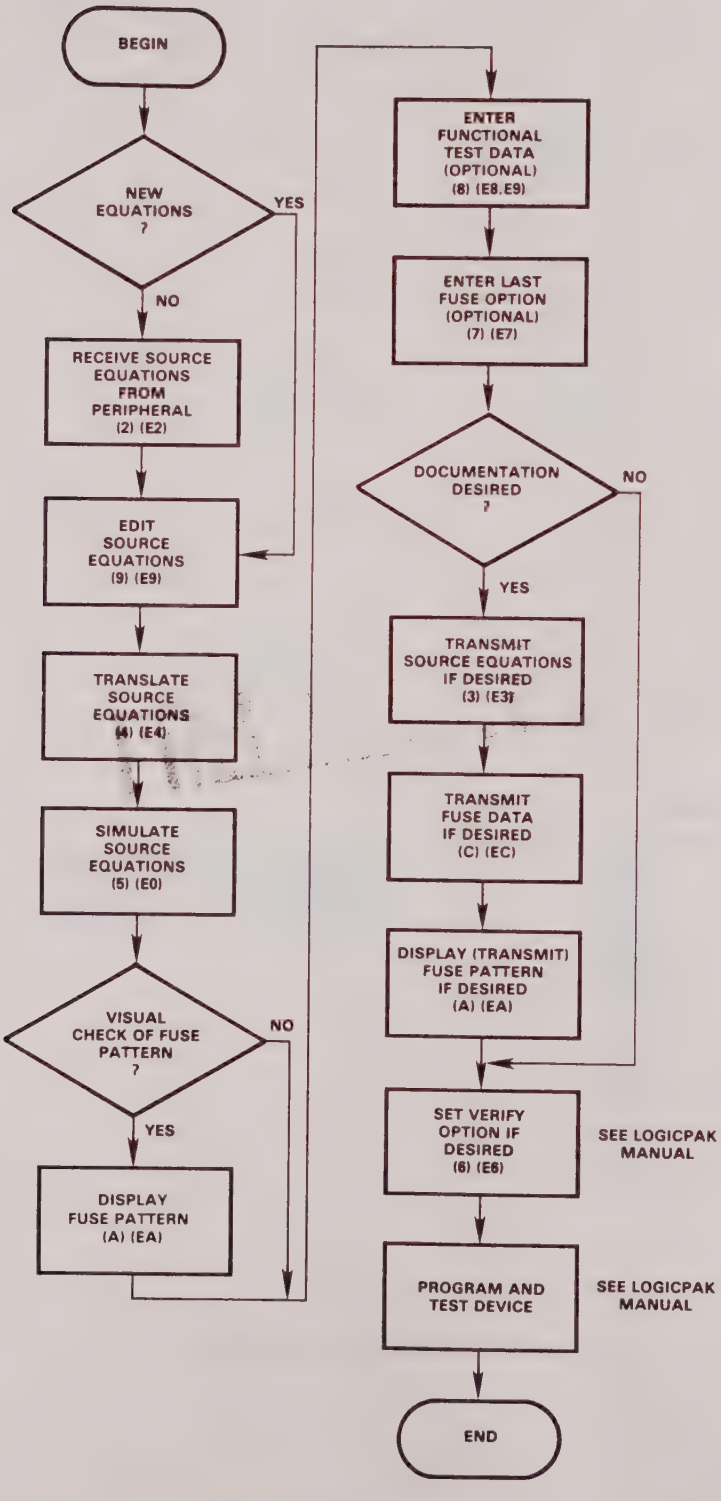
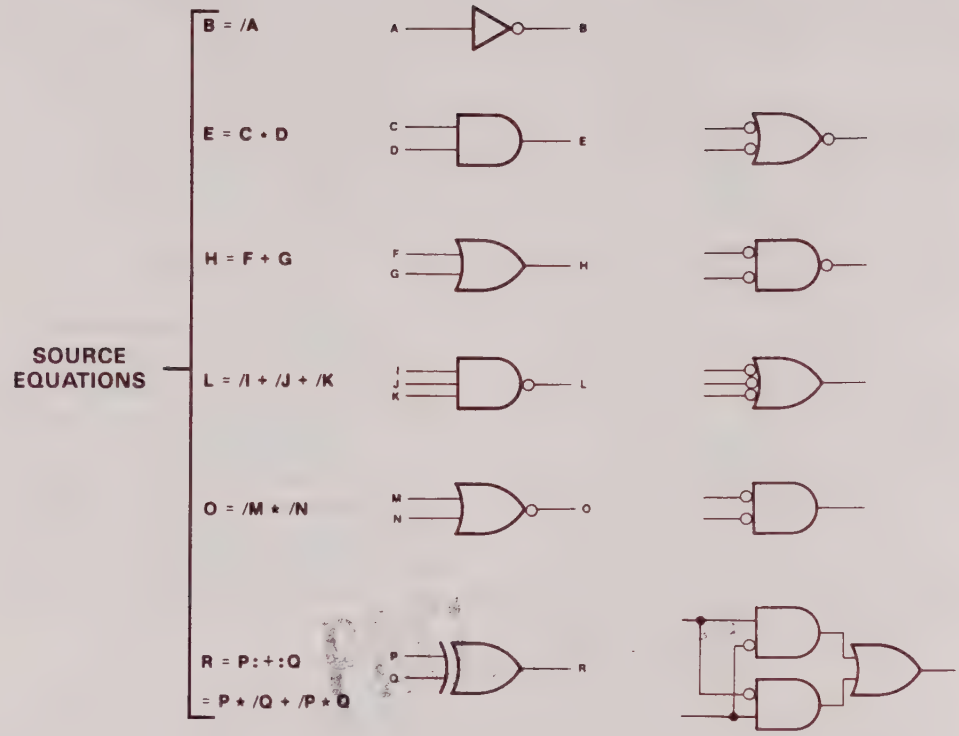

BEGIN
NEW EQUATIONS ?
YES
NO
RECEIVE SOURCE EQUATIONS FROM PERIPHERAL (2) (E2)
EDIT SOURCE EQUATIONS (9) (E9)
TRANSLATE SOURCE EQUATIONS (4) (E4)
SIMULATE SOURCE EQUATIONS (5) (E0)
VISUAL CHECK OF FUSE PATTERN ?
NO
YES
DISPLAY FUSE PATTERN (A) (EA)
ENTER FUNCTIONAL TEST DATA (OPTIONAL) (8) (E8,E9)
ENTER LAST FUSE OPTION (OPTIONAL) (7) (E7)
DOCUMENTATION DESIRED ?
NO
YES
TRANSMIT SOURCE EQUATIONS IF DESIRED (3) (E3)
TRANSMIT FUSE DATA IF DESIRED (C) (EC)
DISPLAY (TRANSMIT) FUSE PATTERN IF DESIRED (A) (EA)
SET VERIFY OPTION IF DESIRED (6) (E6)
SEE LOGICPAK MANUAL
PROGRAM AND TEST DEVICE
SEE LOGICPAK MANUAL
END

Figure 3-1. Flowchart, Data Development

Example 1:
Basic Gates



This example demonstrates how fusable logic can implement the basic inverter. AND. OR. NAND. NOR. and exclusive-OR functions. The PAL 12H6 is selected because it has 12 inputs and 6 outputs. For this example, the fuse pattern is generated using PALASM.

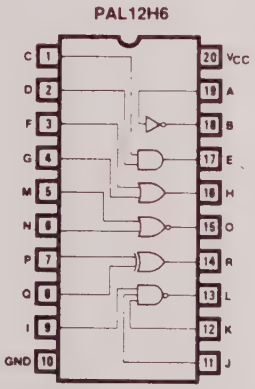


Figure 3-2. Source Equations Example

3.4 THE FUNCTION MENU

When you have completed the setup procedures (section 3.3), the equipment is ready to perform data development. To display the edit menu, press O, or any valid (not a menu item) key. CNTL Z will abort any function on the menu. Escape will take the PLDS out of terminal control.

A set of 15 functions and commands exists for PALASM data development. You should familiarize yourself with them before using the design adapter.

CAUTION

BEFORE REMOVING THE ADAPTER, enter an <esc> from the terminal; or from the Programmer front panel, press the KEYBOARD key (on the System 19) or the VERIFY key (on the 100A or 29A). The processor in the programmer executes firmware resident in the adapter. Precautions must be taken before removing the adapter from the LogicPak to prevent a program interrupt and resulting loss of RAM data.

Figure 3-3 shows how the menu will be displayed on your terminal. Table 3-1 summarizes the purpose of each function and directs you to paragraphs for additional information.

3.5 DATA DEVELOPMENT

This section describes the key sequences (fig. 3-1) used for data development in the order you would most probably follow:

1. Receive or edit source equations.
2. Translate source equations.
3. Compare function tables with source equations and generate Structured Test vectors.
4. Inspect the fuse pattern.
5. Enter data to verify and test the device.

Optional steps include:

6. Select the last fuse option to protect the device from being read.
7. Download the information for inspection or documentation.

Finally:

8. Select the device (instructions in p/t adapter manual).

```
DATA I/O CORPORATION    PLDS-PALASM DESIGN ADAPTER    COPYRIGHT 1982

0 - DISPLAY MENU
1 - ENTER FAMILY PIN CODE
2 - RECEIVE SOURCE EQUATIONS
3 - TRANSMIT SOURCE EQUATIONS
4 - TRANSLATE SOURCE EQUATIONS
5 - SIMULATE SOURCE EQUATIONS
6 - ENTER VERIFY OPTION
7 - ENTER LAST FUSE OPTION
8 - ENTER FUNCTIONAL TEST DATA
9 - EDIT SOURCE EQUATIONS
A - DISPLAY FUSE PATTERN
B - RECEIVE FUSE DATA
C - TRANSMIT FUSE DATA
D - DISPLAY FUSE SUMCHECK
E - ENTER DECIMAL FUSE DATA

<ESC> - BEFORE REMOVING ADAPTER
```

Figure 3-3. Function Menu

Table 3-1. PALASM Menu Commands

CODE		COMMAND	DESCRIPTION	SECTION
FRONT PANEL	TERMINAL			
E1	0	Display Menu	Recalls the menu (fig. 3-3) at any time during the data development process.	
E4	4	Translate Source Equations	Translates the source equations developed with the PALASM Design Adapter (or received from a peripheral source) into data that the programmer will use to program specific fuses.	3.5.4
E0	5	Simulate Source equations	Compares function tables with source equations and generates Structured Test vectors if valid.	3.5.5
N/A	9	Edit Source Equations*	Used to enter and edit source equations and function table.	3.5.3
E2	2	Receive Source Equations	Prepares the programmer to receive source equations from a peripheral via the serial port.	3.5.1
E3	3	Transmit Source Equations	Causes the programmer to output the source equations in RAM, via the serial port, to another peripheral.	3.5.11
EB	B	Receive Fuse Data	Prepares the programmer to receive the fuse pattern of a device, in the JEDEC format, via the serial port of the programmer.	3.5.2
EC	C	Transmit Fuse Data	Outputs fuse pattern, in the JEDEC format, to the programmer serial port.	3.5.12
EA	A	Display Fuse Pattern	Causes the fuse pattern of the selected device to be displayed on the terminal.	3.5.6
E8 E9	8 -	Enter Functional Test Data " "	Allows you to enter data for Logic Fingerprint™ and Structured Test.	3.5.8
N/A	1	Enter Family Pinout Code	Lets you specify a device by entering Family and Pinout Codes from the Data I/O Comparison Chart of Programmable Devices. Codes are automatically specified by the device table when using the design adapter. If second-source devices are used from another manufacturer, its Family & Pinout Codes must be entered manually, after the p/t adapter is installed. The PALASM design adapter will accept only MMI codes.	
E6	6	Enter Verify Option	Lets you select fuse Verify, Structured Test, and Logic Fingerprint™ options. Also affects the automatic verify cycle after programming. The options are:	3.5.9

			Ø = Both fuse Verify and Logic Fingerprint™ if enabled (and Structured Test if data is present in RAM). This is the default. 1 = Fuse Verify only. 2 = Logic Fingerprint™ and Structured Test only. For additional information on fuse Verify options consult the LogicPak Manual.	
E7	7	Enter Last Fuse Option	Enables the programming of special fuses, such as the protect fuse of PAL's. The default is to disable programming. The options are: Ø = Last fuse programming disabled and set fuse date to Ø. 1 = Last fuse programming disabled and set last fuse data to 1. 2 = Last fuse programming enabled and set last fuse data to Ø. 3 = Last fuse programming enabled and set last fuse data to 1.	3.5.10
ED	D	Display Fuse Sum-Check	Displays the sum-check of the fuse states in RAM up to the word limit of the selected device. Does not include test data or source equations.	
EE	E	Enter Decimal Fuse Data	Allows you to key in fuse states by fuse number from the programmer front panel.	3.5.7

Enter And Edit Equations. If you are developing data for the first time, you need to use the edit mode. If you are updating equations previously stored on another media, or entering them after they have been partially developed on a host computer, you will need to use the receive mode.

3.5.1 RECEIVE SOURCE EQUATIONS (2).

This function prepares the programmer to receive source equations from a peripheral via the serial port. The programmer must be prepared to receive data before the transmitting unit begins to send data.

When equations are sent to the programmer in a PLDS configuration using a terminal, it will require a "y" connection or a terminal with a transparent port, so that both terminal and sending unit can access the programmer serial port.

The key sequence is as follows;

1. Complete the preoperational procedure given in section 3.3. The Function menu will be displayed.
2. Enter 2.
3. The programmer is now ready to receive data. The transmitting unit can now be told to send data.

Figure 3-4 shows how source equations would appear on a terminal display after they have been sent to the PLDS. The display is explained in section 3.5.3. All line feeds (<LF>) are stripped from the data in input. Carriage returns (< CR>) are inserted every 80 characters unless they occur sooner. The data is sequentially stored in the source buffer.

If the equations were generated by the PLDS, they will end with an ASCII ETX. Upon receiving the ETX, the programmer will terminate the receive source equations mode and await the next command. Without the ETX, the mode must be aborted from the programmer front panel and the EI mode reentered.

3.5.2 RECEIVE FUSE DATA (B) (EB). The procedure to receive fuse data in the JEDEC format is as follows:

1. Complete any necessary preoperational procedures (sec. 3.1).
2. Enter B.
3. The programmer is now ready to receive data from a peripheral. The transmitting unit may begin sending data in the JEDEC format.

All (<LF>) are stripped from the data on input. (<CR> is inserted every 80 characters unless it occurs sooner. The data is sequentially stored in the source buffer.

Figure 3-5 shows an example of fuse data transmission. For an explanation of the JEDEC format, consult the LogicPak manual section on Input/Output Select functions.

3.5.3 EDIT SOURCE EQUATIONS (9)

This function is used for entering and editing Boolean equations. And is the recommended method for entering Structured Test vectors--via function table entry.

Table 3-2 lists the command set used for entering and editing source equations. The commands will be entered from a terminal and appear as in figure 3-6.

3.5.4 TRANSLATE SOURCE EQUATIONS (4)(E4).

This function translates source equations into a binary fuse pattern that is loaded into the programmer RAM. The procedure is as follows:

1. Enter source equations.
2. Enter 4. The display will show the number of fuses to be blown.
3. The display will also indicate any errors in the equations or the pin list. Section 3.7 lists PALASM error codes.

Data can now be programmed into a device if no testing is required. Additional Select codes are provided for inspection, testing, and editing data after it has been translated.

```
COMMAND = 3 TRANSMIT SOURCE EQUATIONS
PAL12H6                              PAL DESIGN SPECIFICATION
P0055A                               MAX  7/23/82
BASIC GATES EXAMPLE

C D F G M N P Q I GND J K L R O H E B A VCC


B=/A
E=C*B
H=F+G
O=/M*/N
R=P*/Q+/P*Q
L=/I+/J+/K
;DISCRIPTION: THIS EXAMPLE ILLISTRATES THE USE OF FUSIBLE
              LOGIC TO IMPLEMENT THE BASIC GATES.

FUNCTION TABLE
; PIN LIST
C D F G M N P Q I J K L R O H E B A
X X X X X X X X X X X X X X X L H
X X X X X X X X X X X X X X X H L
H H X X X X X X X X X X X X H X X
L H X X X X X X X X X X X X L X X
H L X X X X X X X X X X X X L X X
X X H L X X X X X X X X X X H X X X
X X L H X X X X X X X X X X H X X X
X X X X X X X X L L L H X X X X X X
X X X X L L X X X X X X X H X X X X
X X X X H L X X X X X X X L X X X X
X X X X L H X X X X X X X L X X X X
X X X X X X H L X X X X H X X X X X
X X X X X X L H X X X X H X X X X X
X X X X X X H H X X X X L X X X X X

COMMAND =
```

Figure 3-4. Transmit Source Equations

```
COMMAND = C TRANSMIT FUSE DATA
PAL12H6                                PAL DESIGN SPECIFICATION
P0055A                                 MAX  7/23/82
BASIC GATES EXAMPLE

•B9519•F0
•G1
•L0000
11111110111111111111
1111000000000000000000
0000000000000000000000
0000000000000000000000
0000000000000000001001
11111111111111111111
0000000000000000000000
0000111101111111111111
11111111111111110111
11111111111111111111
•L0200
11101011111111110000
0000000000000000000000
11111111111111011011
11111111111111111110
01111111111111111111
11111111101111111111
11111111111111101111
11111111111111101111
0000000000000000000000
0000
•C1BB9
•V0001 XXXXXXXXXNXXXXXXXL1N
•V0002 XXXXXXXXXNXXXXXXXH0N
•V0003 11XXXXXXXNXXXXXXXHXXN
•V0004 01XXXXXXXNXXXXXXXLXXN
•V0005 10XXXXXXXNXXXXXXXLXXN
•V0006 XX10XXXXXNXXXXXHXXXN
•V0007 XX01XXXXXNXXXXXHXXXN
•V0008 XXXXXXXXXONOOHXXXXXXN
•V0009 XXXX00XXXNXXXXHXXXXN
•V0010 XXXX10XXXNXXXXLXXXXN
•V0011 XXXX01XXXNXXXXLXXXXN
•V0012 XXXXXX10XNXXXHXXXXXN
•V0013 XXXXXX01XNXXXHXXXXXN
•V0014 XXXXXX11XNXXXLXXXXXN
•T01•S0000000000000000000000
•RFFFFFFFF
•E063
COMMAND =
```

Figure 3-5. Transmit Fuse Data

Table 3-2. The Editing Command Set

CHARACTER	NAME	DESCRIPTION
B	Display Line 1	Moves editing pointer to start of buffer and displays first line.
C	Replace Text	Enters change mode; displays <. Character at editing point will be replaced with character entered. Cursor must be beneath a legal character or terminal will beep An error. This will continue until <CR> is reached or until the exit character is entered.
D	Delete Text	Deletes character at editing point from buffer.
E	Display to End	Displays to the end of the buffer and places editing pointer at the end of the buffer.
I	Insert Text	Enters insert mode; moves editing point to the right one space and displays <. Characters and/or lines may be inserted at the current editing point until the exit mode character is entered.
K	Delete Current Line	Deletes line marked by cursor.
L	Display 24 Lines	Displays next 24 lines.
M	Display Commands	Displays the editing command menu.
U	Display Previous Line	Displays last line; places editing pointed at beginning of that line.
SPACE	Move Editing Point Right	Displays next character; moves editing pointer.
RUB/DEL	Rubout	Moves editing pointer left; deletes text during insert.
<BS>	Backspace	Move editing pointer left, but not beyond start of line.
CNTL H	Backspace	Move editing pointer left, but not beyond start of line.
CNTL M	Return	Displays next line.
CNTL P	Purge	Clears buffer.
CNTL Z	Exit Editor	Exits editor, exit "C" & "I" commands. From the "I" or "C" mode, first CNTL Z will return to the edit mode. Second CNTL Z will return to the command mode.
<	Less Than	Identifies that you are in the "I" or "C" mode.

```
COMMAND =  EDIT SOURCE EQUATIONS

B=DISPLAY LINE 1
C=REPLACE TEXT
D=DELETE TEXT
F=DISPLAY TO END
I=INSERT TEXT
K=DELETE CURRENT LINE
L=DISPLAY 24 LINES
M=DISPLAY COMMANDS
U=DISPLAY PREVIOUS LINE
SPACE=MOVE CURSOR RIGHT
RUB/DEL=MOVE CURSOR LEFT/DELETE TEXT DURING INSERT
CONTROL H/BACKSPACE=MOVE CURSOR LEFT
CONTROL M/RETURN=DISPLAY NEXT LINE
CONTROL P=PURGE ALL TEXT
CONTROL Z=EXIT EDITOR/EXIT C & I COMMANDS
```

Figure 3-6. Edit Menu

Figure 3-7 is an example of a PAL design specification. Figure 3-8 is a flow chart of the editing process that follows:

1. Line 0001 must contain the PAL part number left justified followed by PAL DESIGN SPECIFICATION.
2. Line 0002(may contain any optional text) contains the user's part number followed by the originator's name and date.
3. Line 0003(may contain any optional text) lists the device application name.
4. Line 0004(may contain any optional text)is blank. It could contain the user's company name, city and state.
5. Line 0005 must contain the pin list, a sequence of symbolic names (maximum of eight characters) separated by one or more spaces on one or more lines in order of device pin numbers. Each symbolic name must be unqiue. All pins, including Vcc and GND must be specified by the numerical order of the device pins--pin 1 first, Vcc last. Names may be any printable character except the operators "*", "+", "()", ";", or "/". The prefix "/" may be used to logically complement the name.
6. Lines 0006 and 0007 are blank. They may contain comments (optional).
7. Lines 0008 to 0014 contain the equations. The transfer function of the device is expressed in the following three forms:

   a) SYMBOL = EXPRESSION
   b) IF (PRODUCT) SYMBOL = EXPRESSION
   c) SYMBOL: = EXPRESSION

The following terms are used to construct the equations:

SYMBOL	Pin name with optional prefix, "/".
PRODUCT	A sequence of SYMBOLS separated by the AND operator, "*".
IF	Conditional equality, when the PRODUCT is logically true. Otherwise high impedance.
EXPRESSION	A sequence of SYMBOLS separated by operators.
OPERATORS	In hierarchy of evaluation:
	; Comment follows.
	/ Complement, prefix to pin name.
	* AND (product).
	+ OR (sum).
	:+: XOR (exclusive OR).
	:*: XNOR (exclusive NOR).
	() Conditional three-state (IF STATEMENT) or fixed symbol.
	= Equality.
	:= Replaced by after the low-to-high transition the clock.

When entering the Boolean equations, enter the pin identifiers first, and follow each line with a <CR>.

8. Line 0017 contains the words "Function Table" (optional). Enter <Function table> and <CR>.
9. Line 0018 is a comment identifying the pin list (optional).
10. Lines 0019 contains the pin list. Enter the function table pin list using all the same names as the line 0005 pin list. A different order and different polarity is allowed. Each symbolic names is unique, including unused pins. Vcc and GND cannot be listed. Follow the pin list with a <CR>.
11. Enter the function table test vectors, one per line. A vector is a sequence of input and output states listed in the same order as the pin list. It can be followed by an optional comment. Key in CNTL Z when finished.

Valid Function Table States include:

H = High Level.
L = Low Level.
X = Irrelevant.
C = Transition from low to high level.
Z = Off (high impedance).

NOTE

The maximum number of characters that can be entered on any line is 71. Comments can be inserted by beginning the line with a ";" anytime prior to the first test vector.

With active low devices, the pin list and the Boolean equations must be of opposite polarity for output pins only. Some versions of PALASM require a line of dashes before and after the function table test vectors. Some versions also require that the Boolean equation section be terminated by the word "description". These can be included here, but they will be ignored. No comments or the word "description" may follow the function table test vectors.

```
0001  PAL12H6                                PAL DESIGN SPECIFICATION
0002  P0055A                                 MAX  7/23/82
0003  BASIC GATES EXAMPLE
0004
0005  C D F G M N P Q I GND J K L R O H E B A VCC
0006
0007
0008  B=/A
0009  E=C*D
0010  H=F+G
0011  O=/M*/N
0012  R=P*/Q+/P*Q
0013  L=/I+/J+/K
0014  ;DISCRIPTION: THIS EXAMPLE ILLISTRATES THE USE OF FUSIBLE
0015                LOGIC TO IMPLEMENT THE BASIC GATES.
0016
0017  FUNCTION TABLE
0018  ; PIN LIST
0019  C D F G M N P Q I J K L R O H E B A
0020  X X X X X X X X X X X X X X X L H
0021  X X X X X X X X X X X X X X X H L
0022  H H X X X X X X X X X X X X H X X
0023  L H X X X X X X X X X X X X L X X
0024  H L X X X X X X X X X X X X L X X
0025  X X H L X X X X X X X X X H X X X
0026  X X L H X X X X X X X X X H X X X
0027  X X X X X X X X L L L H X X X X X
0028  X X X X L L X X X X X X X H X X X
0029  X X X X H L X X X X X X X L X X X
0030  X X X X L H X X X X X X X L X X X
0031  X X X X X X H L X X X X H X X X X
0032  X X X X X X L H X X X X H X X X X
0033  X X X X X X H H X X X X L X X X X
0034
0035
```

Figure 3-7. Enter Source Equations Example

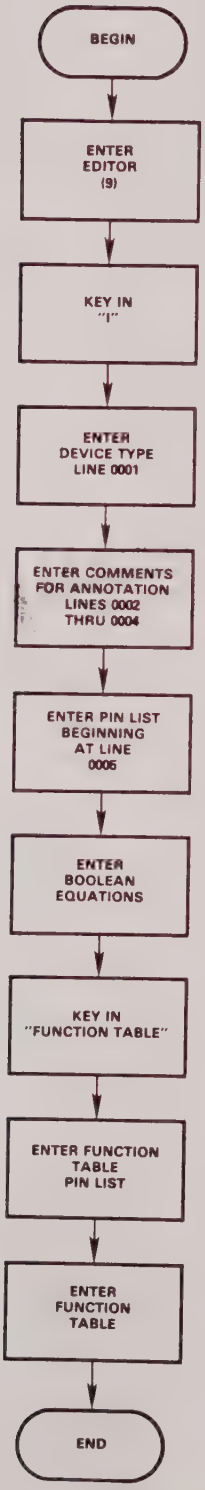


Figure 3-8. Flowchart, Edit Source Equations

3.5.5 SIMULATE SOURCE EQUATIONS (5).

This function compares function tables to source equations. If the two correspond, it generates Structured Test vectors to test the device. The procedure is as follows:

1. Enter source equations and function tables and translate them as described in paragraph 3.5.3.

2. Enter 5.

3. The terminal will display the valid test vectors as shown in figure 3-9. Any invalid inputs and outputs will be displayed. If editing is necessary use function 9.

Simulation Errors. The simulation will detect and prompt the following errors:

1. Illegal Character, Vector Ignore--Function table contains an illegal character. Check function table and replace any illegal characters.
2. Unrecognized Symbol, Aborted--A character used in the funciton table pin list could not be found in the pin list on line 0005. Make sure that you have used the same symbol for the same pin in both lists.
3. A logical function error will be displayed when source equations and function tables do not correspond. For example, "H NOT FOUND ON PIN 19" would be displayed if the source equations required an "H" on pin 19 and the function table was entered with an "L" on pin 19. A logical function error may result from improper entry or invalid equations and function tables. Equations, function tables, and operator entry must all be suspect and should be examined.

3.5.6 DISPLAY FUSE PATTERN(A)

By comparing the logic diagram with the fuse table (fig. 3-10 and fig. 3-11), you can see that the fuse table represents a hard copy of the information in the logic diagram. Fuse numbers have been added to make this comparison easier.

To display the fuse pattern of a device enter "A" from the terminal after completing the preoperational procedures. To examine the pattern on a terminal, it is necessary to halt the display readout. CNTL S halts; CNTL Q continues. The fuse pattern must be a result of either input from a peripheral or equations that have been translated into a fuse pattern.

3.5.7 ENTER DECIMAL FUSE DATA (E).

If it is necessary to enter decimal fuse data in order to alter specific fuse states, this can be done at this time. Data I/O does not recommend this procedure, however, because (1) the resulting fuse pattern will no longer represent the Boolean equations and (2) this will negate any Structured Test vectors.

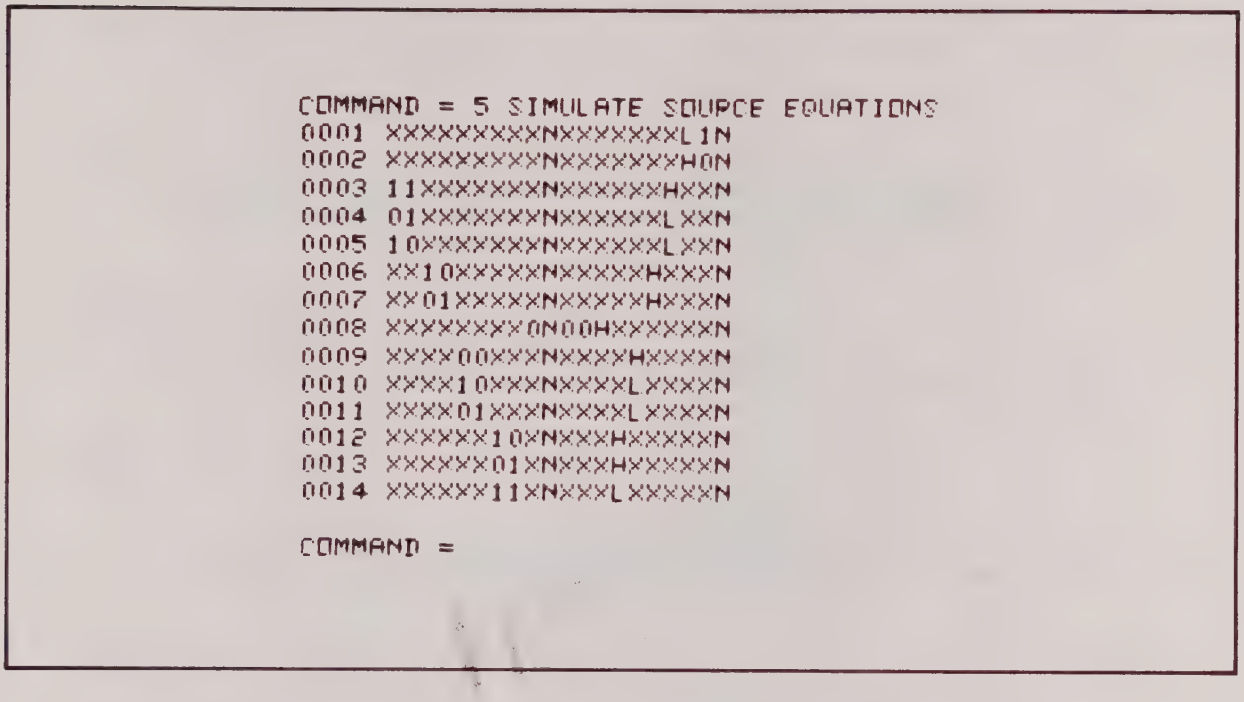


Figure 3-9. Simulate Source Equations

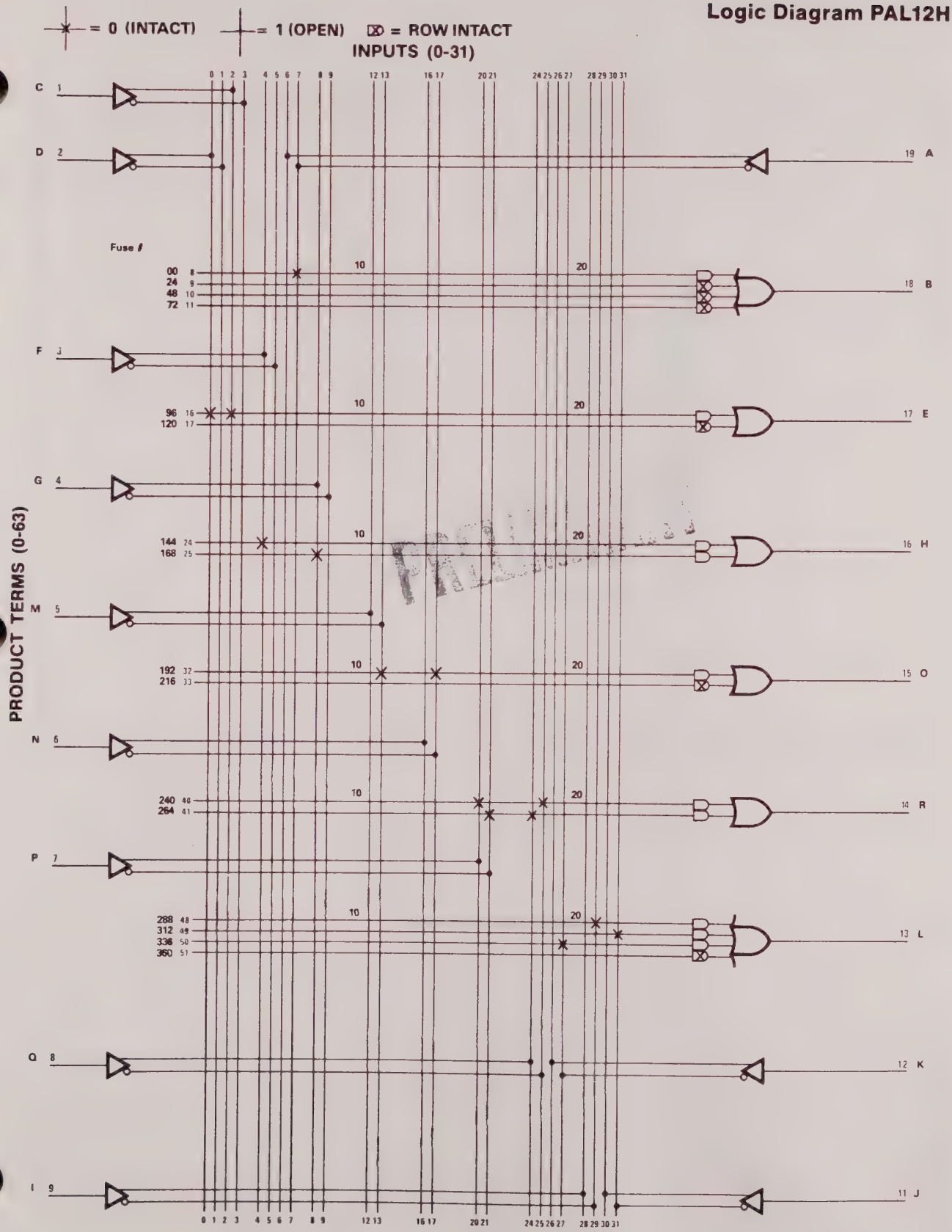
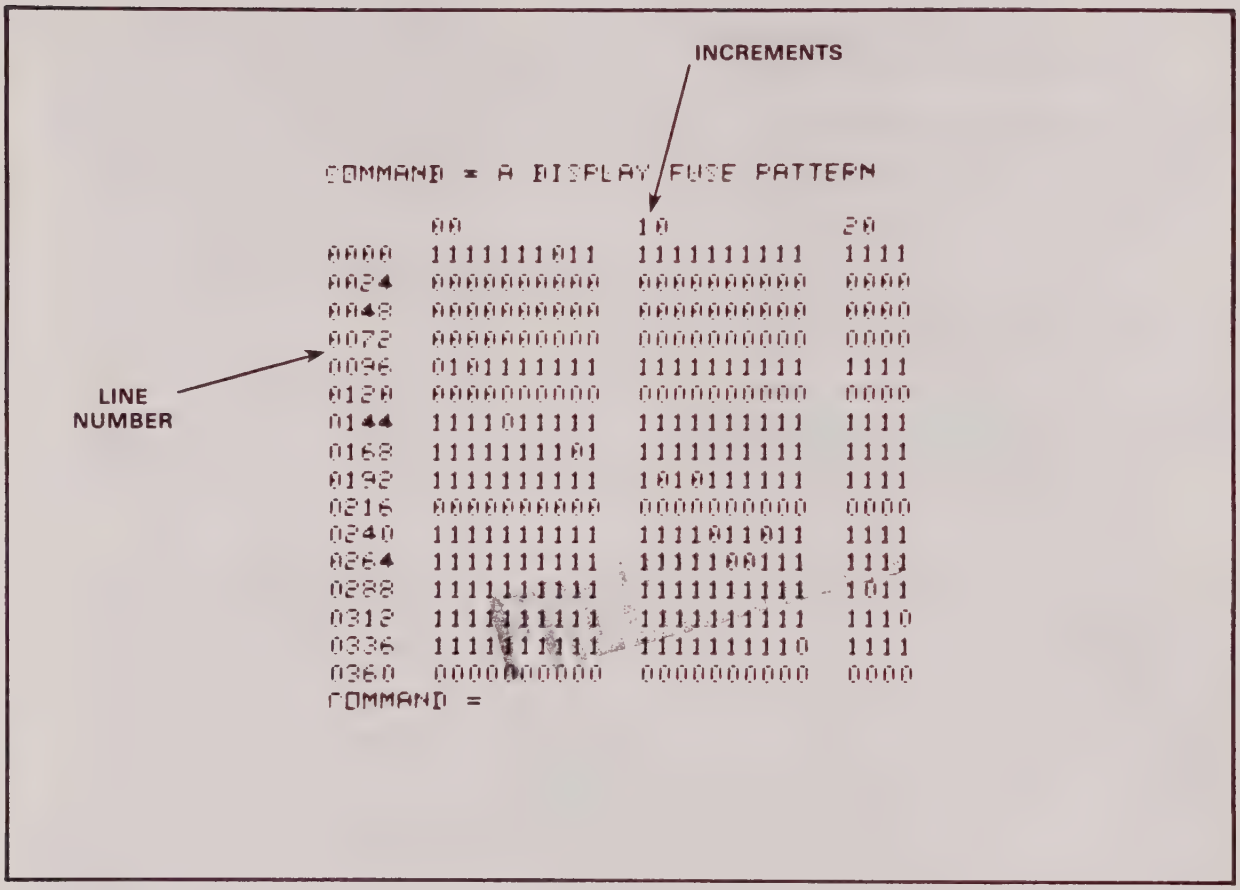


Figure 3-10. Logic Diagram



Line Number + Increment =
Fuse Number

Fuse State
0 = intact
1 = open

Figure 3-11. Fuse Pattern

3.5.8 ENTER FUNCTIONAL TEST DATA (8).

This function lets you view or change the parameters for the Logic Fingerprint Test and the Structured Test. Data entry for these two tests is described in the LogicPak manual section "Verify and Test Select Functions".

3.5.9 VERIFY OPTION (6).

Consult the LogicPak manual section "Verify and Test Select Functions" for information on the Verify option.

3.5.10 ENTER LAST FUSE OPTION (7).

When this function is selected, the terminal displays the default Ø. If another last fuse option is desired, enter the appropriate code from table 3-2.

3.5.11 TRANSMIT SOURCE EQUATIONS (3).

This function tells the programmer to send the source equations stored in RAM to a peripheral via the serial port. It is used mainly for documentation.

Figure 3-4 shows an example of a source equation transmission. It begins with an STX and ends with an ETX. Because all <LF's> are stripped when the data is received (sec. 3.5.1.), every <CR> becomes a <CR>/<LF> on output. The buffer is then dumped to the port. The key sequence is as follows:

1. Complete all necessary setup procedures (sec. 3.3).
2. Set up the unit that is to receive the data.
3. Enter 3.
4. The programmer will now dump the source equations to the serial port.

3.5.12 TRANSMIT FUSE DATA (C).

The procedure for this function is:

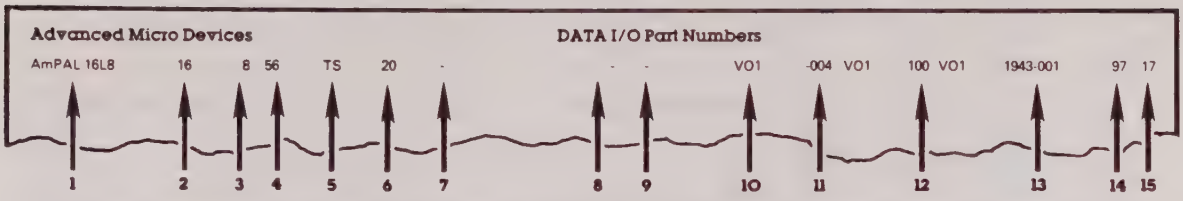
1. Complete any setup procedures (sec. 3.3).
2. Prepare the unit that is to receive the fuse pattern.
3. Enter C.
4. The programmer will now send data to the serial port in the JEDEC format.

Figure 3-5 shows an example of a fuse data transmission. For additional information on the JEDEC format, consult the LogicPak manual section "Input-Output Select functions".

3.6 PALASM ERROR CODES

ERROR	ACTION
Output Pin Undefined	Indicates that you are using a label in the equations that is not part of the pin list. Redefine the output pin.
Colon Error	Indicates an error in the use of a colon (:) with the EX-OR symbol (:+:). Make sure colons (:) are entered correctly.
Equate Error	Firmware expects an equal sign (=) and it is missing from the equations. Check equations and insert equal sign (=) where missing.
Register Error	Firmware expects a colon/equal sign (:=) in the equations and it missing. Check equations and insert colon/equal sign (:=) where missing.
Illegal Device Error	You have entered a name that is not a PAL device. Reenter the correct device name.
EX-OR Error	Firmware expects EX-OR sign (:+:) in the equations and it is missing. Check equations and insert EX-OR sign (:+:) where missing.
Fixed Symbol Error	Arithmetic PAL's contain fixed input combinations. Make sure you have identified the pins correctly that correspond to these fixed inputs.
Unrecognized Data Error	Indicates an unrecognized character in the equations. Check equations for valid characters and reenter where necessary.
Phase Error	Indicates that you have not specified correctly the phase of the output with the pin list, according to the type of the device.
Parenthesis Error	Indicates imbalance of parenthesis in equations. Add or subtract parenthesis where necessary to correct equations.
Too Many Terms Error	The number of products exceeds the specifications of the device. Correct the number of products.
Input Pin Undefined	Same as output pin error except for the input pin.
Pin Undefined	Indicates you haven't defined all the pins on the device. Check pin list and define all pins.

How to use the Comparison Chart: · Programmable Logic Devices



1. **Device Designation** — The device designation represents the manufacturer's designation. Refer to manufacturer's literature for more specific information.

2. **Input Variable** — This column refers to the number of array inputs. Some devices have programmable I/O pins. This number includes these pins as inputs.

3. **Output Variable** — This column refers to the number of outputs. Some devices have programmable I/O pins. This number includes these pins as outputs.

4. **Product Terms** — This column refers to the number of product terms available.

5. **Logic Level** — TS = Tri State AL = Active Low
   OC = Open Collector AH = Active High

6. **Pinout** — Indicates the number of pins of the device leadframe

7. **Programming Module (909/919-XXXX)** — Programming Paks for System 19 and System 17 (e.g., 919-1427)
   Program Card Sets for Models 1, 2, 3, 5, 7, 9 (e.g., 909-1427)
   Interchangeable programming modules have been tested to meet rigid vendor specifications, and have been approved in writing by the semiconductor manufacturers.
   ** = Under development

8. **Revision Level** — The revision level specifies the earliest version of the programming module that will program the device to its manufacturer's latest programming specifications. All later revision programming modules may be used as well. Older programming modules can generally be updated to new specifications in the field through modification and/or recalibration. Contact a Data I/O Service Center for update information.

9. **Socket Adapters (715-XXXX)** — Socket adapters match programming module outputs to device pinout, word limit and byte size requirements (e.g., 715-1428-1). For Flat Pak socket adapters contact your Data I/O representative. Note: some early model Data I/O programmers may require retrofit of a 709-1012 shielded interface cable between the programming module and socket adapter. Contact a Data I/O Service Center if assistance is needed in cable identification.

10. **LogicPak (303A)** — The LogicPak is a universal programming and development module for all programmable logic devices. The "V" number in this column refers to the revision level required to support this device.

11. **Program/Test Adapter (303A-XXX)** — A Program/Test Adapter contains software to configure the LogicPak for an individual device family. The P/T adapters will be updated as new family members are introduced by the manufacturers. The dash number in this column refers to the P/T adapter required to program this device. The "V" number refers to the revision required to support this device.

12. **Design Adapter (303A-XXX)** — The design adapter adds data development capability to the LogicPak. The dash number in this column refers to the design adapter required to support this device. The "V" number refers to the revision required to support this device.

13. **Design Diskette (324-XXXX-XXX)** — Adds data development capabilities to the Model 200A Data Control Unit.

14. **Family Code** - The family code configures the LogicPak for a specific device family.

15. **Pinout Code** — The pinout code configures the P/T adapter to a specific device pinout.

NOTES:

** = Under development
OC = Open Collector
TS = Tri State
FPGA = Field Programmable Gate Array
FPLA = Field Programmable Logic Array
FPLS = Field Programmable Logic Sequencer
FPRP = Field Programmable ROM Patch
I/O PORT = Programmable I/O Port
PAL* = Programmable Array Logic
PMUX = Programmable Multiplexer
† = Part no longer in production

*PAL is a registered trademark of Monolithic Memories Inc.

Device	Input Variable	Output Functions	Product Terms	Logic Level	Pinout	Rev.	Socket Adapter	LogicPak	P/T Adapter	Design Adapter	Design Diskette	Family	Pinout
Advanced Micro Devices						**DATA I/O Part Numbers**							
AmPAL 16L8	16	8	56	TS	20	-	-	VO1	-004 VO1	100 VO1	1943-001	97	17
AmPAL 16R8	16	8	56	TS	20	-	-	VO1	-004 VO1	100 VO1	1943-001	97	24
AmPAL 16R6	16	8	62	TS	20	-	-	VO1	-004 VO1	100 VO1	1943-001	97	24
AmPAL 16R4	16	8	60	TS	20	-	-	VO1	-004 VO1	100 VO1	1943-001	97	24
AmPAL 16LD8	16	8	64	AL	20	-	-	VO1	-004 VO1	**	**	97	17
AmPAL 16H8	16	8	56	TS	20	-	-	VO1	-004 VO1	**	**	97	25
AmPAL 16HD8	16	8	64	AH	20	-	-	VO1	-004 VO1	**	**	97	25
Fairchild Semiconductor						**DATA I/O Part Numbers**							
FPLA 93458 459	16	8	48	OC/TS	28	919-1549 (S19/17 Only)	VO1	1594-1	-	-	-	-	-
Harris Semiconductor						**DATA I/O Part Numbers**							
HPL77153 82S153	8	10	32	TS	20	-	-	VO1	**	**	**	**	**
HPL77209 16L8	16	8	56	TS	20	-	-	VO1	**	**	**	**	**
HPL77210 16R4	16	18	60	TS	20	-	-	VO1	**	**	**	**	**
HPL77211 16R6	16	18	62	TS	20	-	-	VO1	**	**	**	**	**
HPL77212 16R8	16	8	56	TS	20	-	-	VO1	**	**	**	**	**
HPL77215 16H8	16	8	56	TS	20	-	-	VO1	**	**	**	**	**
HPL77216 16Z8	16	8	56	TS	20	-	-	VO1	**	**	**	**	**
Monolithic Memories						**DATA I/O Part Numbers**							
PAL 10H8	10	8		TS	20	E	1428-1(5)	VO1	-002 VO1	100 VO1	1943-001	95	18
PAL 12H6	12	6		TS	20	E	1428-1(5)	VO1	-002 VO1	100 VO1	1943-001	95	19
PAL 14H4	14	4		TS	20	E	1428-1(5)	VO1	-002 VO1	100 VO1	1943-001	95	20
PAL 16H2	16	2		TS	20	E	1428-1(5)	VO1	002 VO1	100 VO1	1943-001	95	22
PAL 10L8	10	8		TS	20	E	1428-2(5)	VO1	002 VO1	100 VO1	1943-001	95	13
PAL 12L6	12	6		TS	20	E	1428-2(5)	VO1	-002 VO1	100 VO1	1943-001	95	14
PAL 14L4	14	4		TS	20	E	1428-2(5)	VO1	002 VO1	100 VO1	1943-001	95	15
PAL 16L2	16	2		TS	20	E	1428-2(5)	VO1	002 VO1	100 VO1	1943-001	95	16
PAL 16A4	16	4		TS	20	E	1428-2(5)	VO1	002 VO1	100 VO1	1943-001	95	24
PAL 16X4	16	4		TS	20	E	1428-2(5)	VO1	002 VO1	100 VO1	1943-001	95	24
PAL 16R4	16	4		TS	20	E	1428-2(5)	VO1	002 VO1	100 VO1	1943-001	95	24
PAL 16R6	16	6		TS	20	E	1428-2(5)	VO1	-002 VO1	100 VO1	1943-001	95	24
PAL 16L8	16	8		TS	20	E	1428-2(5)	VO1	-002 VO1	100 VO1	1943-001	95	17
PAL 16R8	16	8		TS	20	E	1428-2(5)	VO1	002 VO1	100 VO1	1943-001	95	24
PAL 16C1	16	2		TS	20	E	1428-3(5)	VO1	002 VO1	100 VO1	1943-001	95	21
PAL 12H10	12	10		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	07
PAL 14H8	14	8		TS	24	-		VO1	002 VO1	100 VO1	1943-001	95	08
PAL 16H6	16	6		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	09
PAL 18H4	18	4		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	10
PAL 20H2	20	2		TS	24	-		VO1	002 VO1	100 VO1	1943-001	95	11
PAL 12L10	12	10		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	01
PAL 14L8	14	8		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	02
PAL 16L6	16	6		TS	24	-		VO1	002 VO1	100 VO1	1943-001	95	03
PAL 18L4	18	4		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	04
PAL 20L2	20	2		TS	24	-		VO1	-002 VO1	100 VO1	1943 001	95	05
PAL 20C1	20	1		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	06
PAL 20L10	20	10		TS	24	-		VO1	-002 VO1	100 VO1	1943 001	95	06
PAL 20X10	20	10		TS	24	-		VO1	002 VO1	100 VO1	1943-001	95	06
PAL 20X8	20	8		TS	24	-		VO1	002 VO1	100 VO1	1943-001	95	06
PAL 20X4	20	4		TS	24	-		VO1	002 VO1	100 VO1	1943-001	95	06

(5) Reference Field Bulletin 021-0195

National Semiconductor						**DATA I/O Part Numbers**							
PAL 10H8	10	8		TS	20	E	1428 1(5)	VO1	-002 VO1	100 VO1	1943-001	95	18
PAL 12H6	12	6		TS	20	E	1428-1(5)	VO1	-002 VO1	100 VO1	1943-001	95	19
PAL 14H4	14	4		TS	20	E	1428-1(5)	VO1	-002 VO1	100 VO1	1943-001	95	20
PAL 16H2	16	2		TS	20	E	1428-1(5)	VO1	002 VO1	100 VO1	1943-001	95	22
PAL 10L8	10	8		TS	20	E	1428-2(5)	VO1	-002 VO1	100 VO1	1943-001	95	13
PAL 12L6	12	6		TS	20	E	1428-2(5)	VO1	-002 VO1	100 VO1	1943-001	95	14
PAL 14L4	14	4		TS	20	E	1428 2(5)	VO1	-002 VO1	100 VO1	1943-001	95	15
PAL 16L2	16	2		TS	20	E	1428 2(5)	VO1	-002 VO1	100 VO1	1943 001	95	16
PAL 16A4	16	4		TS	20	E	1428-2(5)	VO1	-002 VO1	100 VO1	1943-001	95	24
PAL 16X4	16	4		TS	20	E	1428-2(5)	VO1	-002 VO1	100 VO1	1943-001	95	24
PAL 16R4	16	4		TS	20	E	1428-2(5)	VO1	-002 VO1	100 VO1	1943-001	95	24
PAL 16R6	16	6		TS	20	E	1428-2(5)	VO1	-002 VO1	100 VO1	1943-001	95	24
PAL 16L8	16	8		TS	20	E	1428-2(5)	VO1	-002 VO1	100 VO1	1943-001	95	17
PAL 16R8	16	8		TS	20	E	1428-2(5)	VO1	-002 VO1	100 VO1	1943-001	95	24
PAL 16C1	16	2		TS	20	E	1428-3(5)	VO1	-002 VO1	100 VO1	1943-001	95	21
PAL 12H10	12	10		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	07
PAL 14H8	14	8		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	08
PAL 16H6	16	6		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	09
PAL 18H4	18	4		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	10
PAL 20H2	20	2		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	11
PAL 12L10	12	10		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	01
PAL 14L8	14	8		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	02
PAL 16L6	16	6		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	03
PAL 18L4	18	4		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	04
PAL 20L2	20	2		TS	24	-		VO1	002 VO1	100 VO1	1943-001	95	05
PAL 20C1	20	1		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	06
PAL 20L10	20	10		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	06
PAL 20X10	20	10		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	06
PAL 20X8	20	8		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	06
PAL 20X4	20	4		TS	24	-		VO1	-002 VO1	100 VO1	1943-001	95	06

(5) Reference Field Bulletin 021-0195

Device	Input Variable	Output Functions	Product Terms	Logic Level	Pinout
Raytheon					
PMUX 29693	10	8	4		20
Signetics					
FPLA 82S101 100	16	8	48	OC TS	28
FPLS 82S104 105	16	8	48	OC TS	28
FPRP 82S106 107	16	9	48	OC TS	28
FPGA 82S102 103	16	9		OC TS	28
FPLA 82S152 153	8	10	32	OC	20
FPLS 82S158 159	4	12	32	OC TS	20
8T32	I O PORT	Input Low			24
8T33	I O PORT	Input Low			24
8T35	I O PORT	Input Low			24
8T36	I O PORT	Input Low			24
Texas Instruments					
PAL 16L8	16	8		TS	20
PAL 16R4	16	4		TS	20
PAL 16R6	16	8		TS	20
PAL 16R8	16	8		TS	20
FPLS 74FP335 333	12	6	32	OC TS	24
FPLA 74FP840 839	14	6	32	OC TS	24

Rev	Socket Adapter	LogicPak	P/T Adapter	Design Adapter	Design Diskette	Family	Pinout
DATA I/O Part Numbers							
G	1396						
DATA I/O Part Numbers							
A	1594-1	VO1	-001 VO1	-101 VO1	1944-001	96	01
A	1594 1						
A	1594 2	VO1	-001 VO1	-101 VO1	1944-001	96	03
A	1594 2						
A	1594-1	VO1	-001 VO1	-101 VO1	1944-001	96	04
A	1594 1						
A	1595	VO1	001 VO1	101 VO1	1944-001	96	02
A	1595						
A	0018 001	VO1	-001 VO1	101 VO1	1944 001	96	05
A	0018 001						
	••	••	••	••	••	••	••
A	1608						
A	1608						
A	1608						
A	1608						
DATA I/O Part Numbers							
		VO1	••	••	••	••	••
		VO1	••	••	••	••	••
		VO1	••	••	••	••	••
		VO1	••	••	••	••	••
		VO1	••	••	••	••	••
		VO1	••	••	••	••	••

U.S. and Canada Sales Offices and Representatives
Direct Sales and Service Offices

HAWAII, OREGON, WASHINGTON

Data I/O Corporation
10525 Willows Road N.E.
C-46
Redmond, WA 98052
(206) 881-6444
Telex 15-2167
(Sales & Service)

NORTHERN CALIFORNIA

Data I/O Corporation
473 Sapena Court
Suite 4
Santa Clara, CA 95050
(408) 727-0641
Telex 352054
(Sales, Applications & Service)

WESTERN REGION

Data I/O Corporation
2770 South Harbor Blvd.
Suite K
Santa Ana, CA 92704
(714) 662-1182
Telex 910-595-1562
(Sales & Service)

EASTERN REGION ADMINISTRATIVE OFFICES

MAINE, EASTERN MASSACHUSETTS, NEW HAMPSHIRE, VERMONT

Data I/O Corporation
Nashua Road
Route 101A
Amherst, NH 03031
(603) 889-8511 *(Sales)*
(603) 889-8513 *(Applications & Service)*
Telex 943431

SOUTHERN REGION ADMINISTRATIVE OFFICE

Data I/O Corporation
1810 N. Glenville Drive
Suite 108
Richardson, TX 75081
(214) 235-0044
Telex 792474
(Sales & Applications)

Representatives

(Sales Only)

ALASKA

Microprocessor Technology
3537 East 65th Avenue
Anchorage, AK 99507
(907) 344-8080
Telex (090) 25-462

OREGON, EASTERN WASHINGTON

Northwest Test and Measurement
8196 Southwest Hall
Suite 217
Beaverton, OR 97005
(503) 646-9966

ARKANSAS, LOUISIANA, OKLAHOMA, TEXAS

TESTECH, Inc.
1000 E. Campbell Road
Suite 108
Richardson, TX 75081
(214) 644-5010

9219 Katy Freeway
Suite 124
Houston, TX 77024
(713) 467-2192

2525 Wallingwood
Suite 804
Austin, TX 78746
(512) 327-7033

MICHIGAN, EASTERN KENTUCKY, OHIO, WEST VIRGINIA, WESTERN PENNSYLVANIA

Electro Sales Associates
1635 Mardon Drive
Dayton, OH 45432
(513) 426-5551

29200 Vassar Road
Suite 505
Livonia, MI 48152
(313) 474-7320

9816 Portage Road
Portage, MI 49002
(616) 323-2416

851 East 222nd Street
Cleveland, OH 44123
(216) 261-5440

3740 Mount Royal Blvd.
Allison Park, PA 15101
(412) 487-3801

UPSTATE NEW YORK

Martin P. Andrews
523 East Genesee Street
P.O. Box 443
Fayetteville, NY 13066
(315) 637-5291

METROPOLITAN NEW YORK, LONG ISLAND, NORTHERN NEW JERSEY

RTI
300 Northern Blvd.
Great Neck, NY 11021
(516) 829-3770

KANSAS, MISSOURI, NEBRASKA, SOUTHERN ILLINOIS

Midtec Associates, Inc.
7702 Mize Road
DeSoto, KS 66018
(913) 441-6565
Telex 910-743-4129

8460 Lindbergh North
Suite 11
Florissant, MO 63031
(314) 837-5200

ALABAMA, FLORIDA, GEORGIA, MISSISSIPPI, NORTH CAROLINA, SOUTH CAROLINA, TENNESSEE

Pen Tech Associates
1398 Semoran Blvd.
Suite 106
Casselberry, FL 32707
(305) 645-3444

201 S.E. 15th Terrace
Suite K
Deerfield Beach, FL 33441
(305) 421-4989

Cherokee Center
627 Cherokee Street
Suite 21
Marietta, GA 30060
(404) 424-1931

Holiday Office Center
3322 Memorial Pkwy. S.W.
Suite 36
Huntsville, AL 35801
(205) 881-9298

3709 Alliance Drive
Suite C
Greensboro, NC 27407
(919) 852-6000

INDIANA, IOWA, WESTERN KENTUCKY, MINNESOTA, NORTHERN ILLINOIS, NORTH DAKOTA, SOUTH DAKOTA, WISCONSIN

Torkelson Associates
4940 Viking Drive
Minneapolis, MN 55435
(612) 835-2414

108 Wilmot Road
Suite 110
Dearfield, IL 60015
(312) 945-8700

2346 S. Lynhurst
Room B101
Indianapolis, IN 46241
(317) 244-7867

2840 N. Brookfield Road
Brookfield, WI 53005
(414) 784-7736

ARIZONA, COLORADO, IDAHO, MONTANA, NEW MEXICO, NEVADA, UTAH, WYOMING

Zeus Electronics, Inc.
1428 E. Pierson Street
Phoenix, AZ 85014
(602) 263-6022
Telex 910-951-1362

310 Val Verde S.E.
Albuquerque, NM 87108
(505) 842-6633

3333 Quebec Street
Suite 2950
Denver, CO 80207
(303) 321-4246

1849 W. North Temple
Suite B105
Salt Lake City, UT 84116
(801) 534-0500

SOUTHERN NEW JERSEY, DELAWARE, MARYLAND, EASTERN PENNSYLVANIA, VIRGINIA, WASHINGTON D.C.

SCI-REP, Inc.
304 Cooper Center
Pennsauken, NJ 08109
(609) 662-5222

9512A Lee Highway
Fairfax, VA 22031
(703) 385-0600

CONNECTICUT, RHODE ISLAND, WESTERN MASSACHUSETTS

Pat Jenks Associates
52 Washington Avenue
North Haven, CT 06473
(203) 239-6201

114 Cummings Park
Woburn, MA 01801
(617) 938-0488

CANADA

Allan Crawford Associates, Ltd.
6503 Northam Drive
Mississauga, Ontario L4V 1J2
(416) 678-1500
Telex 06 968769

881 Lady Ellen Place
Ottawa, Ontario K1Z 5L3
(613) 722-7682
Telex 053 3600

7018 Cote De Liesse
St. Laurent, P.Q. H4T 1E7
(514) 731-8564
Telex 05-824944

1935-30th Avenue N.E.
Calgary, Alberta T2E 6Z5
(403) 230-1341
Telex 03 821186

3795 William Street
Burnaby, B.C. V5C 3H3
(604) 294-1326
Telex 04 54247

192 Joseph Zatzman Drive
Dartmouth, Nova Scotia B3B 1N4
(902) 463-9360
Telex 019 31604

15043A 118th Avenue
Edmonton, Alberta T5V 1H9
(403) 451-4893

Addresses on this list subject to change without notice.

International Sales Offices and Representatives

Europe

Data I/O Europe
Vondelstraat 50-52
NL-1054 GE Amsterdam
The Netherlands
(20)186855
Telex 16616 DATIO NL

Data I/O Germany GmbH
Bahnhofstrasse 3
D-6453 Seligenstadt
West Germany
(6182)3088/89
Telex 4184962 DATA D

AUSTRIA

Ing. Ernst Steiner
Hummelgasse 14
A-1130 Wien
Austria
(222)827474
Telex 135026 ES A

BELGIUM

Simac Electronics
Rue du Progres 52
BOITE 3
B-1000 Brussels
Belgium
(2)2192451-3
Telex 23662 SIMEIP B

DENMARK

ITT Komponent A/S
Fabriksparken 31
DK-2600 Glostrup
Denmark
(2)451822
Telex 33355 itt dk

FINLAND

Havulinna-Instrumentarium Oy
P.O. Box 357
SF-00101 Helsinki 10
Finland
(0)799711
Telex 124426 havul sf

FRANCE

M.B. Electronique
606, rue Fourny
Zac de Buc
P.O. Box 31
F-78530, Buc, France
(3)9568131
Telex MB 695414 F

GERMANY

Instrumatic Electronics GmbH
Am Kirchenholzel 14
D-8032 Grafelfing (near Munich)
West Germany
(89)852063
Telex 524298 INSTR D

GREECE

Mr. Elias Mammeas
65 Meg. Alexandrou & Athinas Street
P.O. Box 181
Korydallows
Piraeus
(1)4967815/or 818
Telex 213835 LH GR

ISRAEL

R.D.T. Electronics Engineering, Ltd.
46 Sokolov Street
P.O. Box 75
Ramat Hasharon
Israel
(3)483211
Telex 32143 RDT IL

ITALY

Sistrel SPA
Via Giuseppe Armellini 39
I-00143 Roma
Italy
(6)5915551
Telex 680356 SISTREL i

Sistrel SPA
Via P. da Volpedo 59
I-20092 Cinisello Balsamo (Mi)
Italy
(2)6120129 or 6181893
Telex 334643 SISTREL i

NETHERLANDS

Simac Electronics
Veenstraat 20
NL-5503HR Veldhoven
The Netherlands
(40)533725
Telex 51037 SIMAC NL

NORWAY

Teleinstrument A/S
P.O. Box 134
N-1371 Asker
Norway
(2)789460
Telex 72919 TELIN N

PORTUGAL

Decada
Equipamentos de Electronica, Lda.
Rua Pedro Nunes 47-C
P-1000 Lisboa or P-1003 Lisboa Codex
Portugal
(19)574984
Telex 18469 NONIO P

SPAIN

Instrumatic
Alameda Principal 26
Apartado 151
Malaga 5
Spain
(52)213199/213898
Telex 77131 Hafn e

Instrumatic
Juan Hurtado de Mendoza 9
Madrid-16
Spain
(1)2502577/2507278
Telex 46277 Itic e

SWEDEN

Macrotek AB
Vallingbyvagen 212, Box 43
S-162 11 Vallingbyvagen, Sweden
(8)870190
Telex 12543 MATEK S

SWITZERLAND

Instrumatic SA
5-7, rue du Clos
CH-1207 Geneve
Switzerland
(22)360830
Telex 28667 FLUJA CH

Instrumatic AG
Weingartenstrasse 9
CH-8803 Ruschlikon
Switzerland
(1)7241410
Telex 56605 INST CH

TURKEY

Data I/O Europe
Vondelstraat 50-52
NL-1054 GE Amsterdam
The Netherlands
(20)186855
Telex 16616 DATIO NL

UNITED KINGDOM

Microsystem Services
P.O. Box 37
Lincoln Road
Cressex Industrial Estate
High Wycombe
Bucks, HP12 3XJ, England
(494)41661
Telex 837187 MICSYS G

International

Data I/O International
10525 Willows Road N.E.
C-46
Redmond, WA 98052
U.S.A.
(206) 881-6444
Telex 15-2167

AUSTRALIA

CORPORATE OFFICE

Warburton O'Donnell Ltd.
372 Eastern Valley Road
P.O. Box 182
Chatswood, N.S.W., Australia 2067
4073261
Telex Warfran AA21299

SOUTH AUSTRALIA

ADELAIDE, AUSTRALIA

Warburton Franki (Adelaide) Pty. Ltd.
322 Grange Road
Kidman Park
South Australia 5025 Australia
3567333
Telex Warfran AA82579

BRISBANE, QUEENSLAND

Warburton Franki (Brisbane) Pty. Ltd.
13 Chester Street
Fortitude Valley
Queensland 4006 Australia
527255
Telex Warfran AA41052

MELBOURNE, VICTORIA

Warburton Franki (Melbourne) Pty. Ltd.
220 Park Street
South Melbourne
Victoria 3205 Australia
699 4999
Telex Warfran AA31370

PERTH, WESTERN AUSTRALIA

Warburton Franki (Perth)
98-102 Belgravia Street
Belmont, 6104
Western Australia
65 7000
Telex Warfran AA92908

SYDNEY, NEW SOUTH WALES

Warburton Franki Sydney
199 Parramatta Road
Auburn, N.S.W. 2144
Australia
648 1711
Telex Warfran AA22265

TASMANIA

Associated Agencies Pty., Ltd.
P.O. Box 252
43 Albert Road
Moonah, Tasmania 7009
23 1841
Telex ASSAG AA58206

BRAZIL

Cosele
Instrumentos Electronicos Ltda.
Rue da Consolacao, 867
3.° And. - Conj. 32
01301 - Sao Paulo - SP
Brasil
255-1733 or 256-7421 or 231-5548
Telex (011) 30869 CSEL-BR

HONG KONG

Cummings Engineering Co., Ltd.
90 Sung Wong Toi Road, I/F,
To Kwa Wan, Kowloon,
Hong Kong
3-636227
Telex 74015 CUMMH HX

INDIA

Transmarketing Private, Ltd.
Sterling Center
16/2 Dr. Annie Besant Rd.
Bombay 400-18
India
022 391874
Telex 011-5424

JAPAN

Data I/O Japan Company, Ltd.
Sankei Building
1-8, Sarugaku-cho 2-Chome
Chiyoda-Ku, Tokyo 101
Japan
(03) 295-2656
Telex 2225391 DATAIO J

KOREA

Elcom System, Inc.
Yeongdong Box 50
Seoul, Korea
555-5222 or 557-3836
Telex ADUCEL K25227

MALAYSIA

Cadco Technical Services
67A, Jalan Perisai
Taman Sri Tebrau
Johore Bahru, Malaysia
Cable - CADCOKIT - JOHOR BAHRU

MEXICO

Christensen, S.A.
Guillermo Prieto 76-304
Col. San Rafael
Delegacion Cuahutemoc
06470-Mexico, D.F.
546-25-95
546-29-55
Telex 017-75612 Mycome.

NEW ZEALAND

Wellington-New Zealand:
Warburton Franki, Ltd.
42-43 Oxford Terrace
Lower Hutt, New Zealand
693-016
Telex Warfran NZ 3824

Warburton Franki, Ltd.
P.O. Box 9301; Newmarket
142 Broadway
Auckland, New Zealand
504-458
Telex Warfran NZ 3824

PHILIPPINES

Cummings Engineering Co., Ltd.
90 Sung Wong Toi Road, 1/F,
To Kwa Wan, Kowloon
Hong Kong
3-636227
Telex 74015 CUMMH HX

SINGAPORE

GEA Technology PTE., Ltd.
Units 1003 to 1008, Block 3, 10th Floor
PSA Multi-Storey Complex
Pasir Panjang Road
Singapore 0511
2729412
Telex RS 37162 Answerback Geasin

SOUTH AFRICA

Electronic Building Elements
(PTY) Ltd.
P.O. Box 4609
Pine Square, 18th Street
Hazelwood
Pretoria, South Africa 0001
46-9221/7
Telex 3 0181 SA
Telegrams Elbilem

SOUTH & CENTRAL AMERICA

Dexter Computer Products, Inc.
3465 Torrance Blvd.
Suite N
Torrence, CA 90503
213-542-7333
Telex 182360 DXTR TRNC

TAIWAN

COMPAC Microelectronics, Inc.
11th Fl., Walsin Bldg.
#219, Chung-Hsiao E. Road, Sec. 4
P.O. Box 53-479
Taipei, Taiwan R.O.C.
7529911
Telex 24368 COMTWN

THAILAND

Dynamic Supply Engineering R.O.P.
12 Soi Pasara 1, Sukhumvit 63
Bangkok-11, Thailand
Tel. - 3914434, 39228532
Telex 82455 DYNASUP TH

Warranty 90 Days

Data I/O warrants that all equipment sold pursuant to any resultant agreement shall be free from defects in material or workmanship at the time of delivery. Such warranty shall extend for 90 days. Buyer must provide notice to Data I/O within this prescribed warranty period of any defect; if the defect is not the result of improper usage, service, maintenance or installation and equipment has not been otherwise damaged or modified after delivery, Data I/O shall either replace or repair the defective part or parts of equipment or replace the equipment or refund the purchase price at Data I/O's option after return of such equipment by buyer to Data I/O. Shipment to and from Data I/O's facility shall be borne on account of buyer.

(a) Consequential Damages — Data I/O shall not be liable for any incidental or consequential damages incurred as a result of any defect in any equipment sold hereunder, and Data I/O's liability is specifically limited to its obligation described herein to repair or replace a defective part or parts covered by this warranty.

(b) Exclusive Warranty — The warranty set forth herein is the only warranty, oral or written, made by Data I/O and is in lieu of and replaces all other warranties, expressed or implied, including the WARRANTY OF MERCHANTABILITY AND THE WARRANTY OF FITNESS FOR A PARTICULAR PURPOSE.

0299/6K/183

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 41 REDMOND, WA 98052

POSTAGE WILL BE PAID BY ADDRESSEE

DATA I/O
Reader Comments
Applications Manager
10525 Willows Road N.E.
C-46
Redmond, WA 98052

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY CARD

FIRST CLASS PERMIT NO. 41 REDMOND, WA 98052

POSTAGE WILL BE PAID BY ADDRESSEE

DATA I/O
Warranty/Registration
Service Manager
10525 Willows Road N.E.
C-46
Redmond, WA 98052

Reader Comments

The manual's completeness, accuracy, organization, usability, and reliability: _____

_____

Did you find errors in this manual? _____

How can this manual be improved? _____

_____

_____

Additional comments: _____

_____

Name _____ Title _____

Department _____ M/S _____

Company _____

Address _____

City _____ State _____ Zip _____

Phone ( ) _____

Manual or Part No. _____

Warranty/Registration

Please return this card to Data I/O within five days

Did the packaging of this equipment exhibit any outward signs of physical damage? . ☐ YES ☐ NO

Did this equipment arrive intact, without loose parts or cable damage? ..... ☐ YES ☐ NO

Did the equipment operate on power-up? ..... ☐ YES ☐ NO

Did you attain adequate system performance? ..... ☐ YES ☐ NO

Were any electrical adjustments required? ..... ☐ YES ☐ NO

If you required assistance, was a local Data I/O representative contacted? ..... ☐ YES ☐ NO

Comments: _____

_____

Name _____ Title _____

Department _____ M/S _____

Company _____

Address _____

City _____ State _____ Zip _____

Phone ( ) _____

Model or Description _____ Serial or Part No. _____

Date Received _____

PATTERSON
BINDERY
4900 E LACEY BLVD

DATA I/O

10525 Willows Road N.E./C-46 Redmond, Washington 98052 (206) 881-6444